MYRA

Myra EU CAPTCHA

Handbuch

Versionsverfolgung

Version	Datum	Änderungsgrund
V1.6	08/2026	Neue Abschnitte: • Content Security Policy Aktualisierte Abschnitte: • Funktionsweise • Widget einbinden • Integration • HTML und JavaScript • TYPO3 • FAQ • Widget wird nicht erkannt • Glossar
V1.5	08/2026	Neue Abschnitte: • Traefik • Client aus der Spezifikation • Sitekeys verwalten • Statistiken abrufen • Glossar
V1.4	08/2026	Neue Abschnitte: • Zusätzliche Secrets verwalten • Persönliche Daten ändern • Passwort zurücksetzen • Passwortrichtlinie • Passkeys löschen • Konto löschen • Cookie- und Tracking-Einstellungen Aktualisierte Abschnitte: • Integration

Version	Datum	Änderungsgrund
		• Konto erstellen • Anmelden • Passkey erstellen • Reiter Logins
V1.3	08/2026	Neue Abschnitte: • Zugang erhalten • Einrichtung • Plan während der Testphase festlegen Aktualisierte Abschnitte: • Erste Schritte • Testphase
V1.2	08/2026	Neue Abschnitte: • Erste Schritte • Testphase Aktualisierte Abschnitte: • Einführung • Funktionsweise • Integration testen Abschnitte gelöscht: • Systemarchitektur
V1.1	08/2026	Aktualisierte Abschnitte: • Einführung • Funktionsweise • Systemarchitektur • Ersten Sitekey anlegen • Integration testen • Dashboard • Bereich Sitekey-Liste

Version	Datum	Änderungsgrund
		• Configuration • Integration • Permissions • Challenges • Reiter Profile • Reiter Plans • Reiter Subscription History • Reiter Logins • Reiter Delete Account • Help • Sitekey anlegen • Sitekey konfigurieren • WordPress
V1.0	08/2026	Erstfassung

Inhalt

1. Einführung

Myra EU CAPTCHA schützt Websites und APIs vor Missbrauch wie Formular-Spam und Credential Stuffing. Der Dienst wird von der Myra Security GmbH betrieben und in Europa gehostet. Die Prüfung läuft unsichtbar im Hintergrund ab; Besuchende lösen kein Rätsel.

> **Video:** Überblick über das Myra EU CAPTCHA und die Weboberfläche — abspielbar in der Online-Hilfe

1.1 Merkmale

Der Dienst unterscheidet sich in den folgenden Punkten von klassischen CAPTCHA-Verfahren:

Merkmal	Beschreibung
Formular- und Anmeldeschutz	Der Dienst schützt Formulare und Anmeldeseiten vor Spam und Credential Stuffing.
Bot-Schutz	Eine Fingerabdruck-Erkennung erkennt Anfragen von Bots und blockiert sie.
Überprüfung ohne Benutzereingabe	Nutzende lösen kein Rätsel und klicken kein Kontrollkästchen an. Die Verifizierung erfolgt automatisch über kryptografische Berechnungen im Hintergrund.
Intelligente Risikoerkennung	Die Risikoanalyse passt den Schwierigkeitsgrad der Challenges dynamisch an, sobald ein Verhalten auffällig wird.
Signalbasiert	Die Erkennung wird täglich mit mehr als 100 Milliarden CDN-Signalen trainiert.
Ohne Cookies oder Datenspeicherung	Der Dienst arbeitet ohne HTTP-Cookies und ohne dauerhaften Browserspeicher. Ein Cookie-Banner ist dafür nicht erforderlich.
Europäischer Betrieb	Der Dienst wird in Europa betrieben und ist auf die DSGVO ausgelegt.
Selbsteinrichtung	Die Integration erfordert keine Änderung am DNS und ist in drei Schritten abgeschlossen.
Plattformübergreifende Integration	Für CMS-Plattformen, Frontend-Frameworks, Backend-Sprachen, mobile Apps und Infrastruktur-Werkzeuge stehen Bibliotheken und Skripte bereit.
API-Unterstützung	Eine REST-API steht für die programmatische Anbindung zur Verfügung.

Merkmal	Beschreibung
Dediziertes Dashboard	Eine Weboberfläche zeigt die Statistiken und verwaltet die Sitekeys.
Erstellung von Sitekeys	Ein neuer Sitekey entsteht durch die Eingabe der zu schützenden Domain.
Konfiguration von Sitekeys	Der Schutz lässt sich je Sitekey aktivieren oder deaktivieren. Schwierigkeitsgrad, Zeitvorgabe und parallele Challenges sind einstellbar.
Anpassung des Widgets	Für das Widget steht ein Dunkelmodus zur Verfügung.
Zugriffsverwaltung	Teammitglieder erhalten Lese- oder Schreibzugriff auf einen Sitekey.

1.2 Einschränkungen

Das EU CAPTCHA befindet sich in einer frühen Ausbaustufe. Beachten Sie die folgenden Einschränkungen:

Einschränkung	Beschreibung
Keine Integration mit anderen Myra-Produkten	Eine Verknüpfung mit weiteren Produkten von Myra Security ist derzeit nicht verfügbar.
Keine Migration	Eine Migrationsoption aus einer bestehenden CAPTCHA-Lösung ist derzeit nicht verfügbar.
JavaScript erforderlich	Die Verifizierung beruht auf JavaScript und einem clientseitigen Proof-of-Work. Ohne aktiviertes JavaScript funktioniert sie nicht.
Rechenzeit auf schwachen Geräten	Auf leistungsschwachen Geräten beansprucht der Proof-of-Work eine spürbare Rechenzeit.
Kein absoluter Schutz	Auch eine signalbasierte Bot-Erkennung schließt hochentwickelte Bots mit echten Browserumgebungen nicht vollständig aus.

Verstehen Sie die Risikobewertung als eine Schutzschicht, nicht als alleinige Sicherheitsmaßnahme.

1.3 Sitekey und Secret

Jede geschützte Domain erhält ein Schlüsselpaar:

Schlüssel	Format	Verwendung
Sitekey	UUID	Öffentlich. Steht im HTML der Seite und identifiziert die Website.
Secret	Base64	Vertraulich. Wird ausschließlich serverseitig für die Verifikation verwendet.

> Geben Sie das Secret niemals im Browser aus und legen Sie es nicht in einem öffentlichen Repository ab.

Beide Werte stehen im Bereich **Public Sitekey and Secret(s)**. Sie erreichen ihn über die Schaltfläche **Integrations** in der Zeile des Sitekeys, Schritt **Details**. Siehe Ansicht Details.

1.4 Funktionsweise

Zwei Bestandteile erbringen den Schutz: das Widget, das in einem versteckten Iframe auf der geschützten Seite läuft, und die Verifikations-API des EU CAPTCHAs. Die Prüfung läuft im Hintergrund und verlangt von den Besuchenden keine Eingabe.

1.4.1 Bestandteile

Myra EU CAPTCHA besteht aus den folgenden Komponenten:

Komponente	Adresse	Aufgabe
Dashboard	`https://app.eu-captcha.eu`	Verwaltet Konto, Sitekeys, Berechtigungen und Statistiken.
Widget	`https://cdn.eu-captcha.eu/verify.js`	Führt die Prüfung im Browser der Nutzenden aus und erzeugt den Token.
Verifikations-API	`https://api.eu-captcha.eu/v1`	Prüft den Token serverseitig.
Dokumentation	`https://docs.eu-captcha.eu`	Enthält dieses Handbuch.

Die geschützte Seite lädt das Widget aus dem CDN und spricht die Verifikations-API an. Wenn Ihre Website eine Content Security Policy sendet, dann muss diese beide Adressen freigeben. Siehe **Content Security Policy**.

1.4.2 Ablauf einer Prüfung

Eine Prüfung läuft in den folgenden Schritten ab:

1. Die geschützte Seite lädt `verify.js` vom CDN.
2. `verify.js` erzeugt ein verstecktes Iframe und lädt darin `check.html`.
3. `check.html` lädt `check.all.js`. Dieses Skript führt die eigentliche Prüfung aus.
4. Das Iframe meldet den erzeugten Token an `verify.js` zurück.
5. `verify.js` schreibt den Token in ein verstecktes Eingabefeld mit dem Namen `eu-captcha-response`.
6. Beim Absenden des Formulars gelangt der Token an den eigenen Server.
7. Der eigene Server prüft den Token über den Endpunkt `POST /verify` der Verifikations-API.

> Die Prüfung im Browser allein schützt nicht. Erst die serverseitige Prüfung des Tokens entscheidet über Annahme oder Ablehnung einer Anfrage. Siehe **Widget einbinden**.

1.4.3 Risikobewertung

Das versteckte Iframe erhebt beim Laden der Seite einen Fingerabdruck aus dutzenden Signalen des Browsers. Die Besuchenden sind daran nicht beteiligt. Die folgenden Gruppen von Signalen fließen in die Bewertung ein:

Signalgruppe	Beispiele
Hardware	Merkmale des Geräts und die Geometrie des Bildschirms
Rendering	Darstellung über WebGL und Audio
Verhalten	Mausbewegung, Scrollverhalten und Interaktionen
Netzwerk	Merkmale der Verbindung und der Herkunft der Anfrage

Aus diesen Signalen berechnet der Dienst einen Risikowert zwischen 0.0 und 1.0.

Die Gründe für eine Blockierung gibt der Dienst nicht über die API aus. Eine offengelegte Erkennungslogik würde es Angreifenden erlauben, den Schutz gezielt zu umgehen. Siehe **Grund für eine Blockierung nicht sichtbar**.

1.4.4 Challenge und Proof-of-Work

Der Risikowert bestimmt den Schwierigkeitsgrad der Challenge: Je höher das Risiko, desto aufwendiger die Rechenaufgabe. Der Dienst kennt acht Schwierigkeitsgrade:

Schwierigkeitsgrad	Rechenaufwand	Trifft zu auf
0	rund 600 Millisekunden	Regelfall bei menschlichen Besuchenden. Die Prüfung bleibt unbemerkt.
1 bis 6	ansteigend	Auffällige Signale erhöhen den Aufwand schrittweise.
7	rund 30 Minuten	Verkehr mit sehr hohem Risiko.

Die Rechenaufgabe ist ein Proof-of-Work. Der Browser erhält zusammen mit der Aufgabe einen Zielwert und probiert in einem Hintergrund-Thread so lange Werte durch, bis das Ergebnis den Zielwert erreicht. Als Verfahren dient Argon2id. Es ist speicherintensiv und dadurch gegen Angriffe über Grafikkarten und Botnetze widerstandsfähig.

Das Widget meldet die Lösung an die Verifikations-API:

- Antwortet die API mit `wait`, rechnet der Browser weiter.
- Antwortet die API mit `success`, ist die Challenge bestanden und das Widget erhält den signierten Token.

> Eine fehlgeschlagene Challenge kennt der Dienst nicht. Eine Challenge wird entweder gelöst oder sie dauert länger. Eine ungewöhnlich lange Dauer weist auf Verkehr mit hohem Risiko hin. Siehe **Challenge dauert ungewöhnlich lange**.

> Der Anfangswert, den Sie je Sitekey einstellen, umfasst die Stufen 0 bis 3. Die höheren Stufen vergibt die Risikoerkennung selbst. Siehe **Sitekey konfigurieren**.

1.4.5 Prüfmodi

Der Zeitpunkt, zu dem die Prüfung startet, ist einstellbar. Die folgenden Modi stehen zur Verfügung:

Modus	Zeitpunkt der Prüfung	Einsatz
Solved Immediate (Standard)	Die Prüfung startet, sobald die Seite geladen wird.	Bestes Nutzererlebnis: Die Prüfung ist bereits abgeschlossen, wenn die Nutzenden das Formular absenden.
Solved Triggered	Die Prüfung startet erst, wenn die Nutzenden das Formular berühren.	Seiten mit vielen Aufrufen, aber wenigen Absendungen, zum Beispiel Checkout- oder Ticketseiten.

Der Modus wirkt sich auf den Verbrauch aus: Im Modus **Solved Immediate** zählt jeder Seitenaufruf mit Formular als Assessment, im Modus **Solved Triggered** nur die tatsächlich begonnenen Absendungen.

Den Modus stellen Sie in den Einstellungen Ihrer eigenen Anwendung um, also dort, wo das Widget eingebunden ist. Die Ansicht **Sitekey - [Domain]** enthält dafür keine Einstellung.

Ein Assessment zählt, sobald das Widget die Prüfung im Hintergrund beginnt. Die Prüfung im Hintergrund und die zugehörige serverseitige Verifikation zählen zusammen als ein Assessment, unabhängig vom gewählten Modus.

2. Erste Schritte

Dieses Kapitel führt vom leeren Konto zur geschützten Website: Konto anlegen, am Kundenportal anmelden, den ersten Sitekey erzeugen, das Widget einbauen, die serverseitige Prüfung ergänzen und die fertige Integration im Kundenportal testen.

In den nächsten Abschnitten werden die folgenden Themen behandelt:

- **Zugang erhalten** beschreibt die Registrierung eines Kontos und die Anmeldung am Kundenportal.
- **Einrichtung** beschreibt den Weg vom ersten Sitekey über das Widget bis zum bestätigten Integrationstest.
- **Testphase** beschreibt die 30 Tage der kostenlosen Testphase und die Zeit danach.

2.1 Zugang erhalten

Der Zugang zum Kundenportal von Myra EU CAPTCHA setzt ein Benutzerkonto voraus. Neue Nutzende legen es selbst unter `https://app.eu-captcha.eu` an, vorhandene Nutzende melden sich mit ihren Zugangsdaten an.

In den nächsten Abschnitten werden die folgenden Themen behandelt:

- **Konto erstellen** beschreibt die Registrierung mit einer E-Mail-Adresse oder über einen Anmeldedienst.
- **Anmelden** beschreibt die Anmeldung mit Passwort, mit zweitem Faktor und mit Passkey.

Für die Registrierung ist keine Zahlungsangabe nötig. Jedes neue Konto beginnt mit einer Testphase von 30 Tagen. Siehe **Testphase**.

Wie das Passwort eines bestehenden Kontos geändert wird, steht unter **Passwort ändern**.

2.1.1 Konto erstellen

Für die Nutzung des Myra EU CAPTCHA benötigen Sie ein Konto im Kundenportal. Sie legen es selbst an. Eine Kreditkarte ist dafür nicht erforderlich.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben Zugriff auf ein E-Mail-Postfach.

Um ein Konto zu erstellen, gehen Sie wie folgt vor:

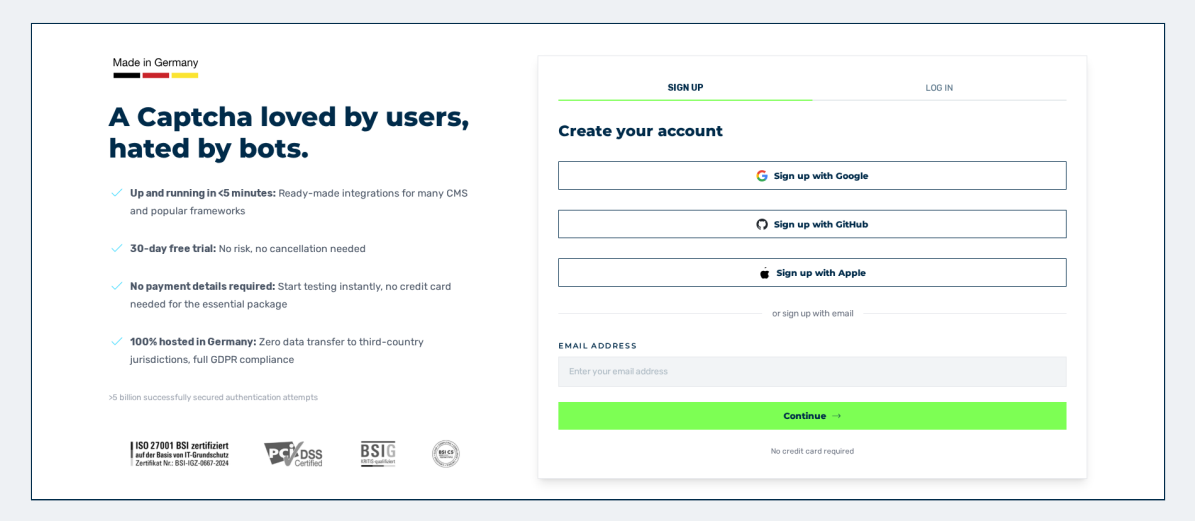


Abbildung 1: Ansicht Create your account

- ► Rufen Sie die Registrierungsseite des Kundenportals auf.
- ► Geben Sie in das Feld **Email address** Ihre E-Mail-Adresse ein.
- ► Klicken Sie auf die Schaltfläche **Continue**.
  - ↳ Die Ansicht **Complete your account** wird angezeigt.

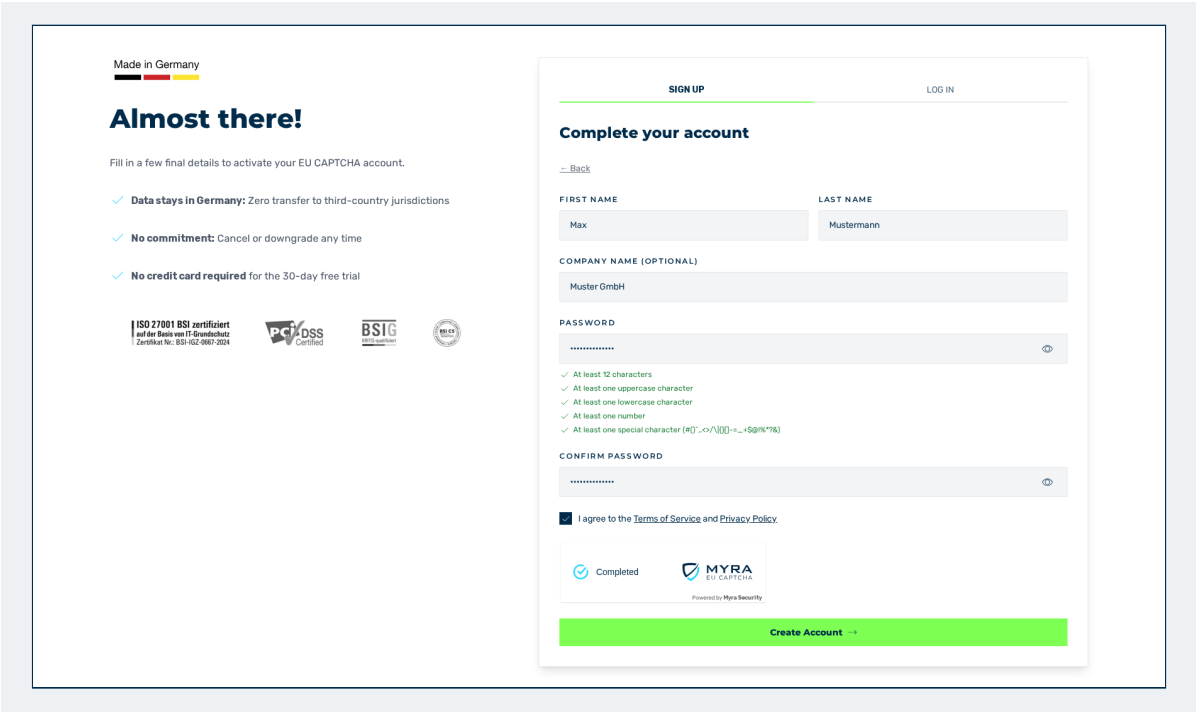


Abbildung 2: Ansicht Complete your account

- ► Geben Sie in die Felder **First name** und **Last name** Ihren Vornamen und Ihren Nachnamen ein.
- ► Falls erforderlich, geben Sie in das Feld **Company name (optional)** den Namen Ihres Unternehmens ein.
- ► Geben Sie in die Felder **Password** und **Confirm password** dasselbe Passwort ein.
  - ↳ Unter dem Feld **Password** zeigt eine Prüfliste an, welche Regeln das Passwort bereits erfüllt.
- ► Aktivieren Sie das Kontrollkästchen vor **I agree to the Terms of Service and Privacy Policy**.
- ► Lösen Sie das Widget **EU CAPTCHA** unter dem Kontrollkästchen.
- ► Klicken Sie auf die Schaltfläche **Create Account**.
  - ↳ Die Ansicht **Verify your email** wird angezeigt. Das System sendet einen sechsstelligen Bestätigungscode an die angegebene E-Mail-Adresse.

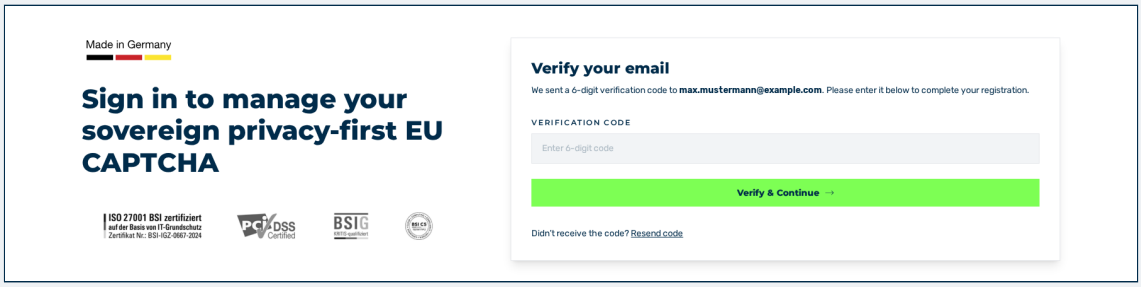


Abbildung 3: Ansicht Verify your email

- ► Geben Sie in das Feld **Verification code** den Code aus der E-Mail ein.
- ► Klicken Sie auf die Schaltfläche **Verify & Continue**.
- ➔ Das Konto ist angelegt. Sie sind angemeldet. Die Ansicht **Dashboard** wird angezeigt.

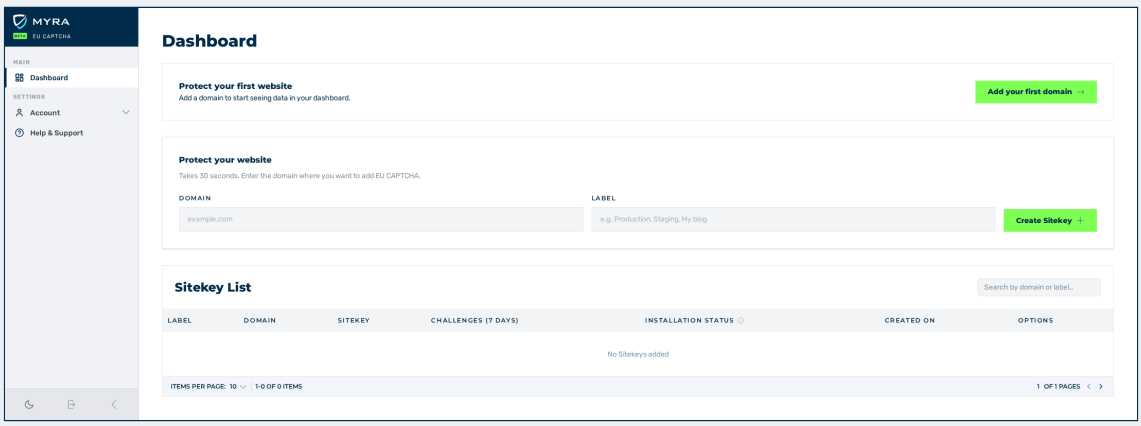


Abbildung 4: Ansicht Dashboard

Über die Schaltfläche ← **Back** kehren Sie von der Ansicht **Complete your account** zur Eingabe der E-Mail-Adresse zurück.

Das Passwort muss die folgenden Regeln erfüllen:

Regel	Bedingung
Länge	Mindestens 12 Zeichen
Großbuchstabe	Mindestens ein Zeichen A bis Z
Kleinbuchstabe	Mindestens ein Zeichen a bis z
Ziffer	Mindestens eine Ziffer 0 bis 9

Regel	Bedingung
Sonderzeichen	Mindestens ein Zeichen aus `#()^,.<>/|{}[]-=_+$@!%*?&`

Wenn für die angegebene E-Mail-Adresse bereits ein Konto besteht, dann zeigt die Ansicht die Meldung **An account with this email already exists.** an. Zusätzlich bietet die Ansicht die Links **Log in** und **Forgot password?** an.

Wenn der Bestätigungscode nicht ankommt, dann fordern Sie ihn über den Link **Resend code** erneut an.

Anstelle von E-Mail-Adresse und Passwort erstellen Sie das Konto über die folgenden Schaltflächen:

- **Sign up with Google**
- **Sign up with GitHub**
- **Sign up with Apple**

Eine Schaltfläche für Passkeys enthält die Ansicht nicht. Ein Passkey lässt sich nur zur Anmeldung verwenden. Siehe **Passkey erstellen**.

Siehe **Anmelden**.

2.1.2 Anmelden

Sie melden sich mit E-Mail-Adresse und Passwort am Kundenportal an. Wenn für Ihr Konto die Zwei-Faktor-Authentifizierung aktiv ist, dann fordert das Portal zusätzlich einen Code an.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben ein Konto erstellt und Ihre E-Mail-Adresse bestätigt. Siehe **Konto erstellen**.

Um sich anzumelden, gehen Sie wie folgt vor:

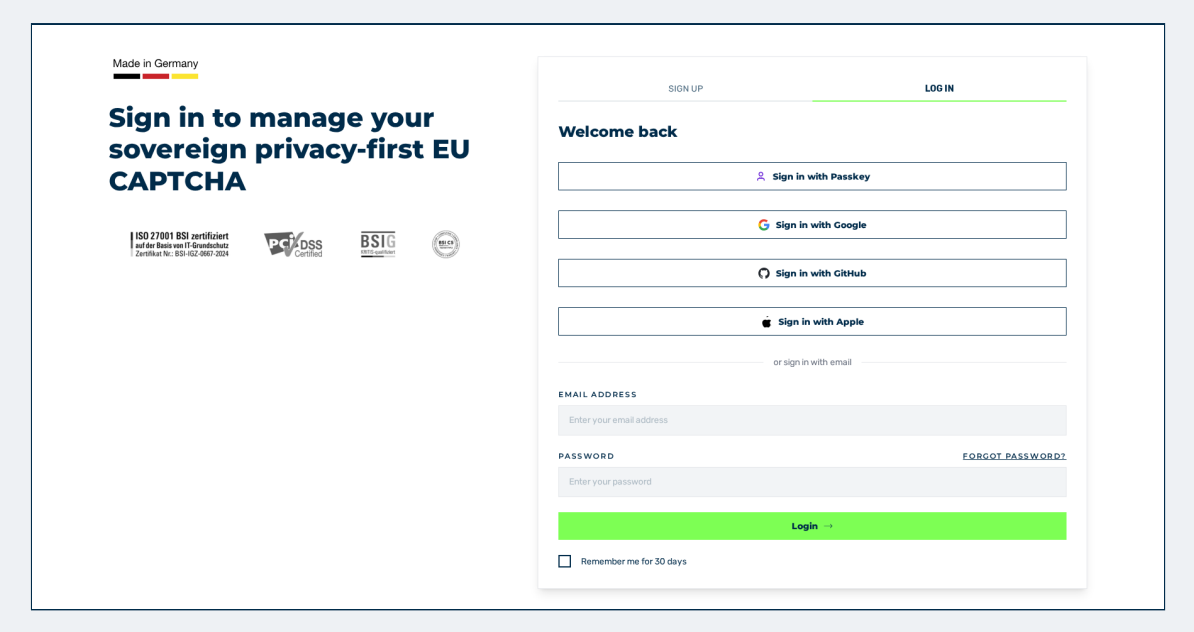


Abbildung 5: Ansicht Welcome back

- ► Rufen Sie die Anmeldeseite des Kundenportals auf.
- ► Geben Sie in das Feld **Email address** Ihre E-Mail-Adresse ein.
- ► Geben Sie in das Feld **Password** Ihr Passwort ein.
- ► Falls erforderlich, aktivieren Sie das Kontrollkästchen **Remember me for 30 days**.
- ► Klicken Sie auf die Schaltfläche **Login**.
  - ↳ Wenn die Zwei-Faktor-Authentifizierung aktiv ist, dann wird die Ansicht **Two-Factor Authentication** angezeigt.

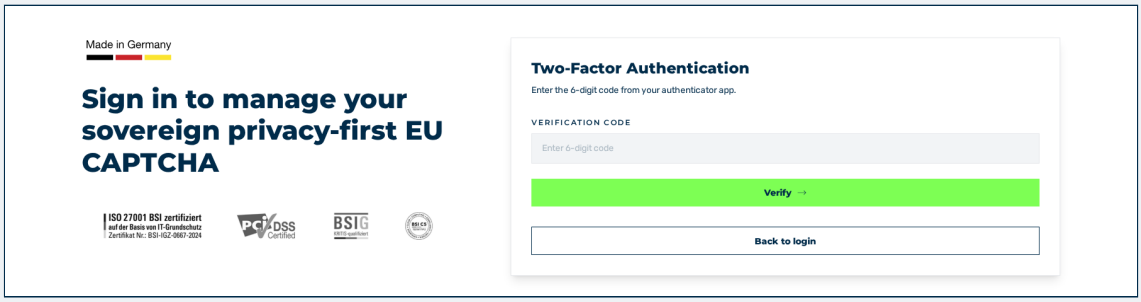


Abbildung 6: Ansicht Two-Factor Authentication

- ► Geben Sie den sechsstelligen Code aus Ihrer Authenticator-App ein.
- ► Klicken Sie auf die Schaltfläche **Verify**.
- ➔ Die Ansicht **Dashboard** wird angezeigt.

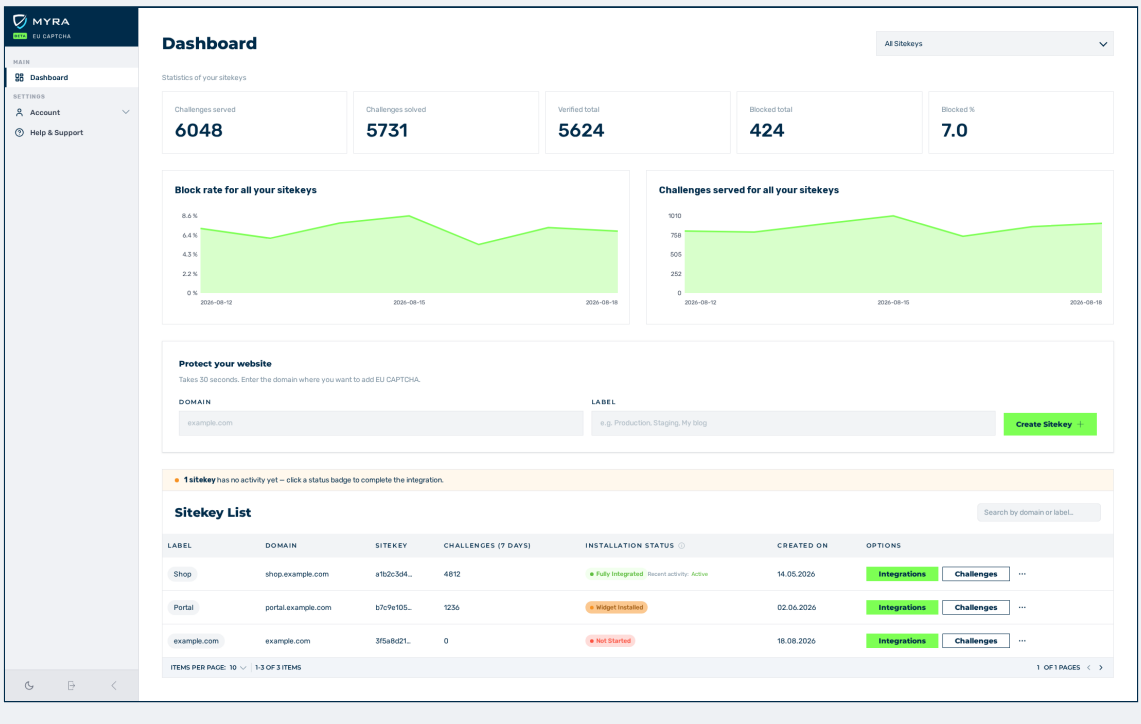


Abbildung 7: Ansicht Dashboard

Über die Schaltfläche **Back to login** kehren Sie von der Ansicht **Two-Factor Authentication** zur Anmeldeseite zurück.

Anstelle von E-Mail-Adresse und Passwort melden Sie sich über die folgenden Schaltflächen an:

- **Sign in with Passkey**
- **Sign in with Google**

- **Sign in with GitHub**
- **Sign in with Apple**

> Die Reihenfolge der Schaltflächen richtet sich nach Ihrem Gerät. Auf Geräten von Apple steht **Apple** an erster Stelle, auf Geräten mit Android steht **Google** an erster Stelle. Auf allen anderen Geräten steht **Passkey** an erster Stelle. Auf Geräten mit Android entfällt die Schaltfläche **Sign in with Apple**.

Wenn Sie Ihr Passwort vergessen haben, dann klicken Sie auf der Anmeldeseite auf den Link **Forgot Password?**. Das System sendet Ihnen eine E-Mail mit einem Link auf die Ansicht **Reset Password**.

Siehe Ansicht Dashboard.

2.2 Einrichtung

Dieser Abschnitt führt vom leeren Konto bis zur bestätigten Integration: den ersten Sitekey anlegen, das Schlüsselpaar abrufen, das Widget einbauen, den Token serverseitig prüfen und das Ergebnis im Kundenportal bestätigen lassen.

Das Kundenportal begleitet diesen Weg mit einem Assistenten. Er führt in drei Schritten durch die Ansichten **Details**, **Integration** und **Integration Test** und zeigt darüber die Angabe **Step ⅓** bis **Step 3/3** an. Siehe **Sitekey**.

In den nächsten Abschnitten werden die folgenden Themen behandelt:

- **Ersten Sitekey anlegen** beschreibt, wie der erste Sitekey für die eigene Domain entsteht und wo das Schlüsselpaar steht.
- **Widget einbinden** beschreibt das Widget im Formular und die Prüfung des Tokens auf dem eigenen Server.
- **Integration testen** beschreibt den Test, der beide Teile der Integration bestätigt.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben ein Konto unter `https://app.eu-captcha.eu` und sind angemeldet. Siehe **Zugang erhalten**.

☑ Sie kennen die Domain, die geschützt werden soll.

☑ Sie haben Zugriff auf den Quelltext der Seite mit dem Formular und auf den Server, der das Formular entgegennimmt.

Fertige Beispiele für gängige Kombinationen aus Frontend und Backend stehen unter **Integration**.

2.2.1 Ersten Sitekey anlegen

Ein Sitekey verbindet eine Domain mit dem Myra EU CAPTCHA. Für jede Domain, die Sie schützen wollen, legen Sie einen eigenen Sitekey an. Der Sitekey besteht aus einem öffentlichen Schlüssel für das Widget und einem Secret für die serverseitige Prüfung.

> **Video:** Anlegen des ersten Sitekeys im Kundenportal — abspielbar in der Online-Hilfe

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet. Siehe **Anmelden**.

☑ Sie kennen die Domain, die Sie schützen wollen.

Um den ersten Sitekey anzulegen, gehen Sie wie folgt vor:

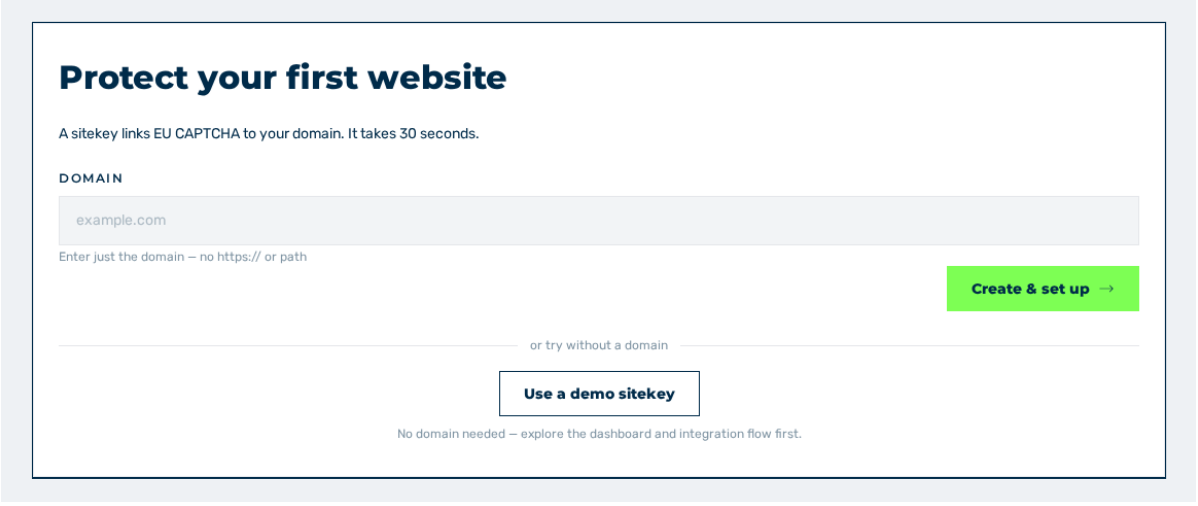


Abbildung 8: Dialog Protect your first website

- ▶ Öffnen Sie die Ansicht **Dashboard**.
  - ↳ Solange kein Sitekey vorhanden ist, wird der Dialog **Protect your first website** angezeigt. Die Beschreibung lautet **A sitekey links EU CAPTCHA to your domain. It takes 30 seconds.**
- ▶ Geben Sie in das Feld **Domain** die zu schützende Domain ein, zum Beispiel `example.com` . Der Hinweis unter dem Feld lautet **Enter just the domain — no https:// or path**.
- ▶ Klicken Sie auf die Schaltfläche **Create & set up**.
  - ↳ Das System legt den Sitekey an und zeigt die Meldung **Success** mit dem Text **You have successfully created your first Sitekey! Now it is time to integrate the Sitekey on your website. We will show you how.** an.

➔ Die Ansicht **Integration** des neuen Sitekeys wird angezeigt.

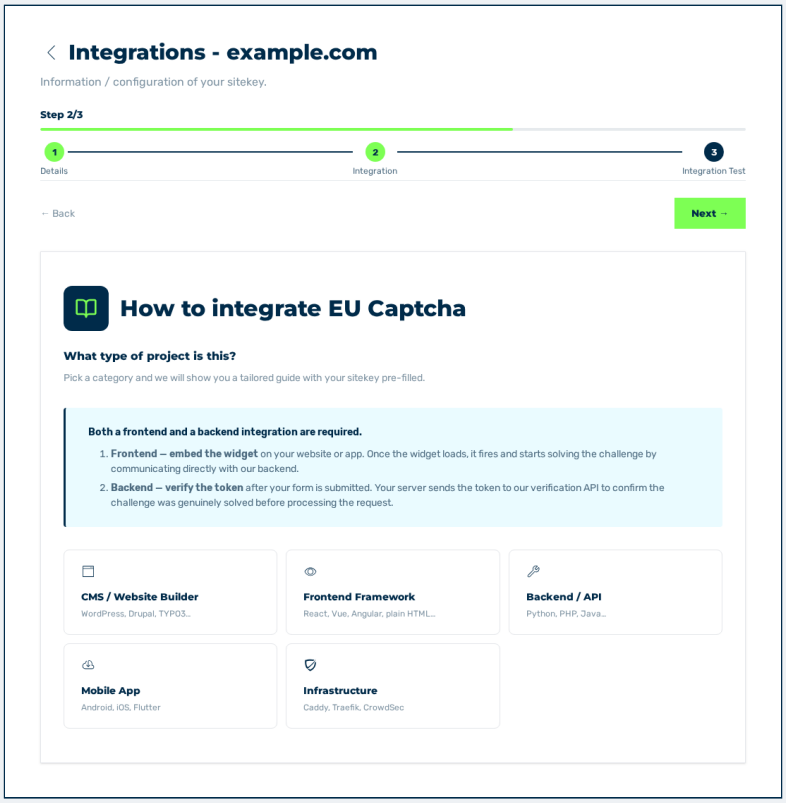


Abbildung 9: Ansicht Integration

> Das System verhindert derzeit nicht, dass mehrere Sitekeys für dieselbe Domain entstehen. Prüfen Sie die Liste der Sitekeys und löschen Sie doppelte Einträge. Siehe **Doppelter Sitekey für dieselbe Domain**.

2.2.1.1 Sitekey ohne eigene Domain

Unter dem Trenner **or try without a domain** legt die Schaltfläche **Use a demo sitekey** einen Sitekey mit einer erzeugten Demo-Domain an. Die Beschreibung lautet **No domain needed — explore the dashboard and integration flow first.**

Das System bleibt in der Ansicht **Dashboard** und zeigt die Meldung **Sitekey created** mit dem Text **Your sitekey is ready. Use the integration guide below to add EU CAPTCHA to your project.** an. Der neue Eintrag wird in der Liste **Sitekey List** hervorgehoben.

2.2.1.2 Dialog erneut öffnen

Wenn Sie den Dialog schließen, zeigt die Ansicht **Dashboard** den Hinweis **Protect your first website** an. Die Beschreibung lautet **Add a domain to start seeing data in your dashboard.**

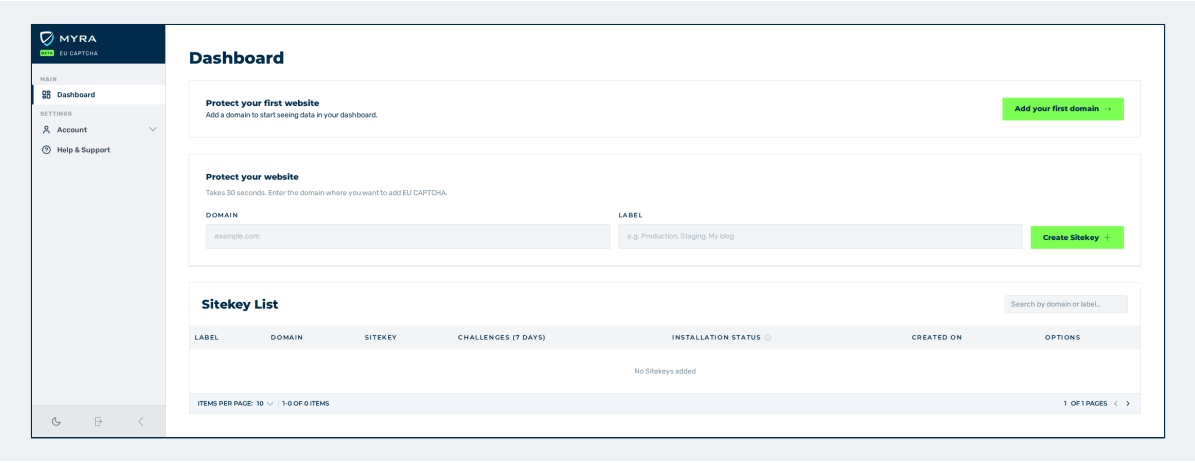


Abbildung 10: Ansicht Dashboard ohne Sitekey

- ► Klicken Sie auf die Schaltfläche **Add your first domain**.
- ➔ Der Dialog **Protect your first website** wird erneut angezeigt.

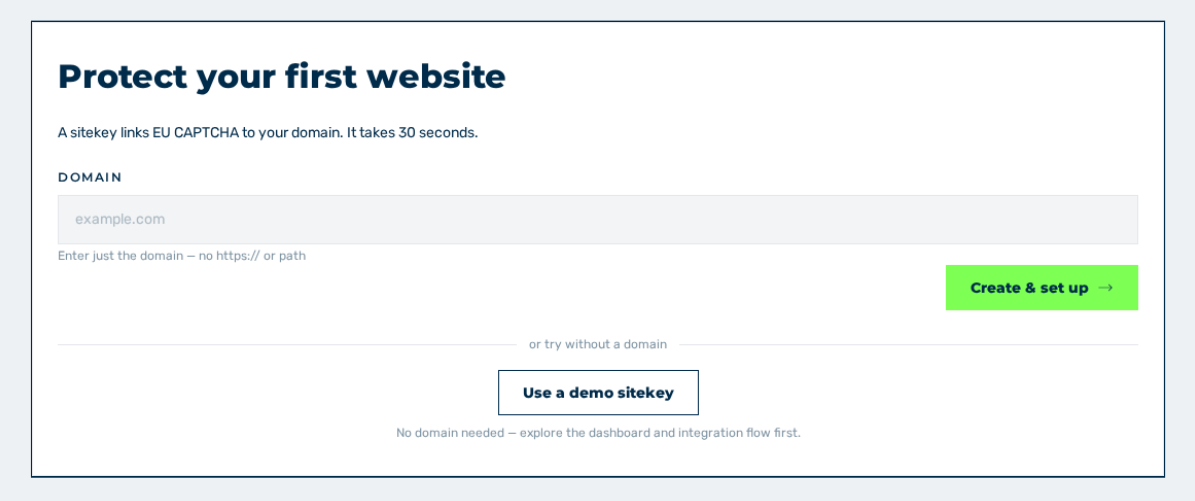


Abbildung 11: Dialog Protect your first website

Siehe **Widget einbinden**.

2.2.2 Widget einbinden

Das Widget wird in drei Schritten eingebunden:

- ein Skript im Kopfbereich der Seite
- ein Element an der Stelle des Logos
- eine Prüfung des Tokens auf Ihrem Server

Änderungen am DNS sind nicht erforderlich.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben einen Sitekey angelegt. Siehe **Ersten Sitekey anlegen**.

☑ Sie kennen den öffentlichen Sitekey und das Secret. Beide stehen in der Ansicht **Details** des Sitekeys. Siehe **Ansicht Details**.

☑ Sie können das HTML und den serverseitigen Quelltext Ihrer Website bearbeiten.

2.2.2.1 Schritt 1: Skript einbinden

Fügen Sie das folgende Element in den Bereich `<head>` Ihrer Seite ein:

```
<script src="https://cdn.eu-captcha.eu/verify.js" async defer></script>
```

Wenn Ihre Website eine Content Security Policy sendet, dann muss diese die Adressen `https://cdn.eu-captcha.eu` und `https://api.eu-captcha.eu` freigeben. Ohne diese Freigabe blockiert der Browser das Widget. Siehe **Content Security Policy**.

2.2.2.2 Schritt 2: Widget-Element einfügen

Fügen Sie das folgende Element an der Stelle ein, an der das Myra-Logo erscheinen soll, üblicherweise innerhalb des zu schützenden Formulars:

```
<div class="eu-captcha" data-sitekey="a1b2c3d4-0000-0000-0000-000000000000"></
div>
```

Ersetzen Sie den Wert von `data-sitekey` durch Ihren öffentlichen Sitekey.

2.2.2.3 Schritt 3: Token serverseitig prüfen

Prüfen Sie das Token auf Ihrem Server, bevor Sie das Formular verarbeiten:

```
POST https://api.eu-captcha.eu/v1/verify
Content-Type: application/json

{
  "sitekey": "a1b2c3d4-0000-0000-0000-000000000000",
  "secret": "a1b2c3d4••••",
  "token": "TOKEN_AUS_DEM_WIDGET"
}
```

Wenn die Antwort den Erfolg bestätigt, dann verarbeiten Sie die Absendung. Andernfalls weisen Sie die Absendung zurück.

Das Secret gehört ausschließlich auf Ihren Server. Geben Sie es niemals im HTML oder in JavaScript aus.

2.2.2.4 Plattformspezifische Anleitungen

Für die verbreiteten Plattformen stehen Bibliotheken und Erweiterungen bereit, die diese drei Schritte übernehmen. Siehe **Integration**.

Siehe **Integration testen**.

2.2.3 Integration testen

Die Ansicht **Integration Test** prüft, ob das Widget auf Ihrer Seite lädt und ob Ihr Server das Token prüft. Der Test benennt beide Teile getrennt, sodass Sie erkennen, welcher Teil noch fehlt.

> **Video:** Prüfen der Integration im Kundenportal — abspielbar in der Online-Hilfe

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben das Widget in Ihre Website eingebunden. Siehe **Widget einbinden**.

☑ Sie haben die geschützte Seite mindestens einmal aufgerufen und das Formular abgesendet.

Um die Integration zu prüfen, gehen Sie wie folgt vor:

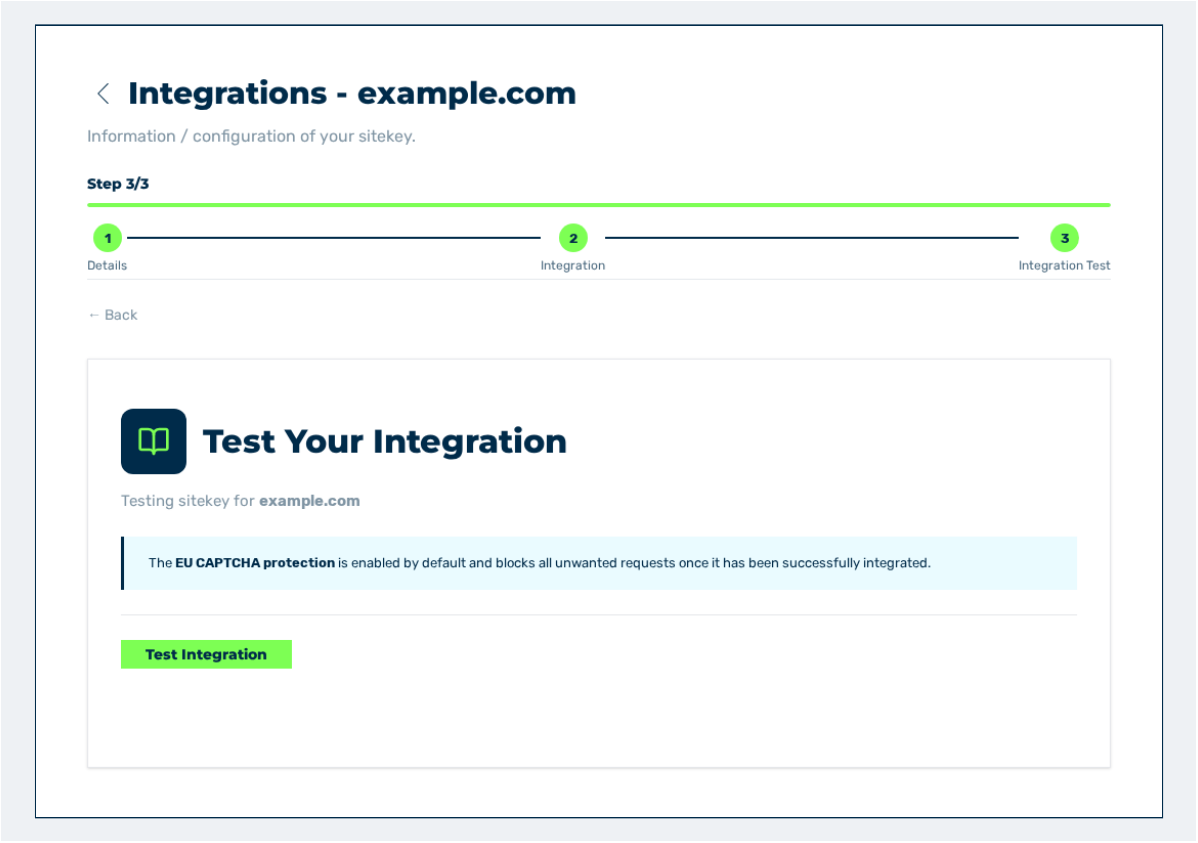


Abbildung 12: Ansicht Integration Test

► Öffnen Sie den Sitekey und wechseln Sie in die Ansicht **Integration Test**.

► Klicken Sie auf die Schaltfläche **Test Integration**.

➔ Das System zeigt das Ergebnis des Tests an.

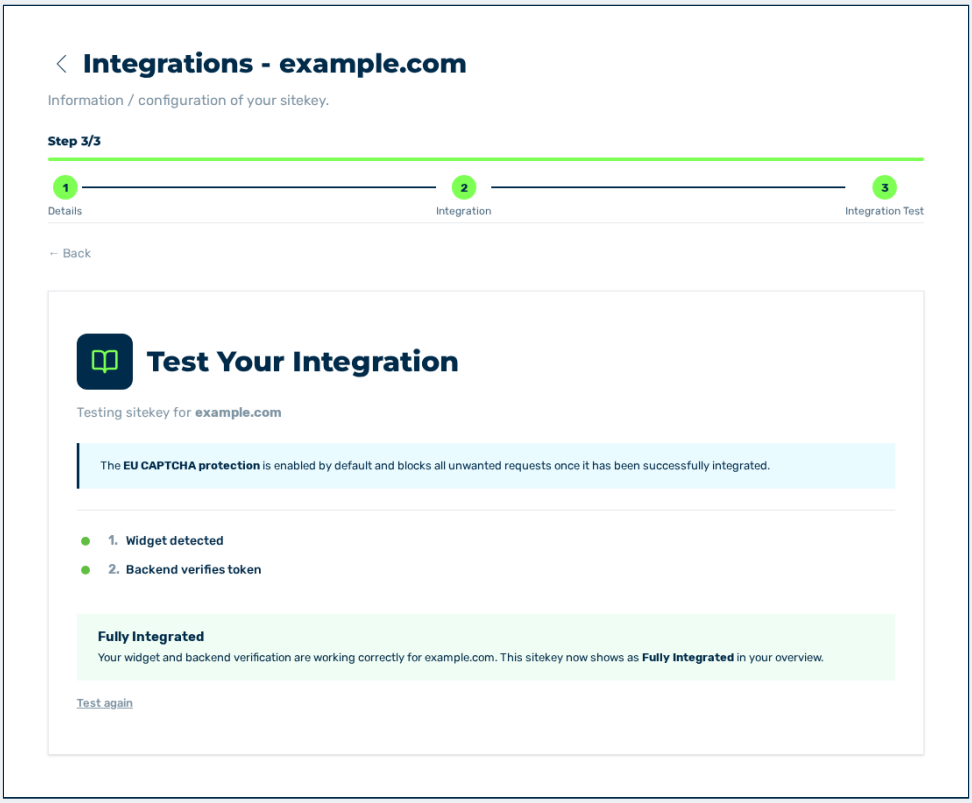


Abbildung 13: Ansicht Integration Test mit dem Ergebnis Fully Integrated

Das Ergebnis nimmt einen der folgenden Werte an:

Ergebnis	Bedeutung	Maßnahme
Fully Integrated	Das Widget lädt, und Ihr Server prüft das Token.	Keine. Die Integration ist abgeschlossen.
Widget is working but your server didn't verify the token.	Das Widget lädt, die serverseitige Prüfung fehlt.	Ergänzen Sie den Aufruf des Endpunkts `/verify` . Siehe **Widget einbinden**.
Widget not detected.	Das Widget lädt nicht.	Prüfen Sie das Skript-Element und das Element mit der Klasse `eu-captcha` .

2.2.3.1 Funktionsprüfung im Betrieb

Prüfen Sie zusätzlich den Ablauf aus Sicht eines Besuchers:

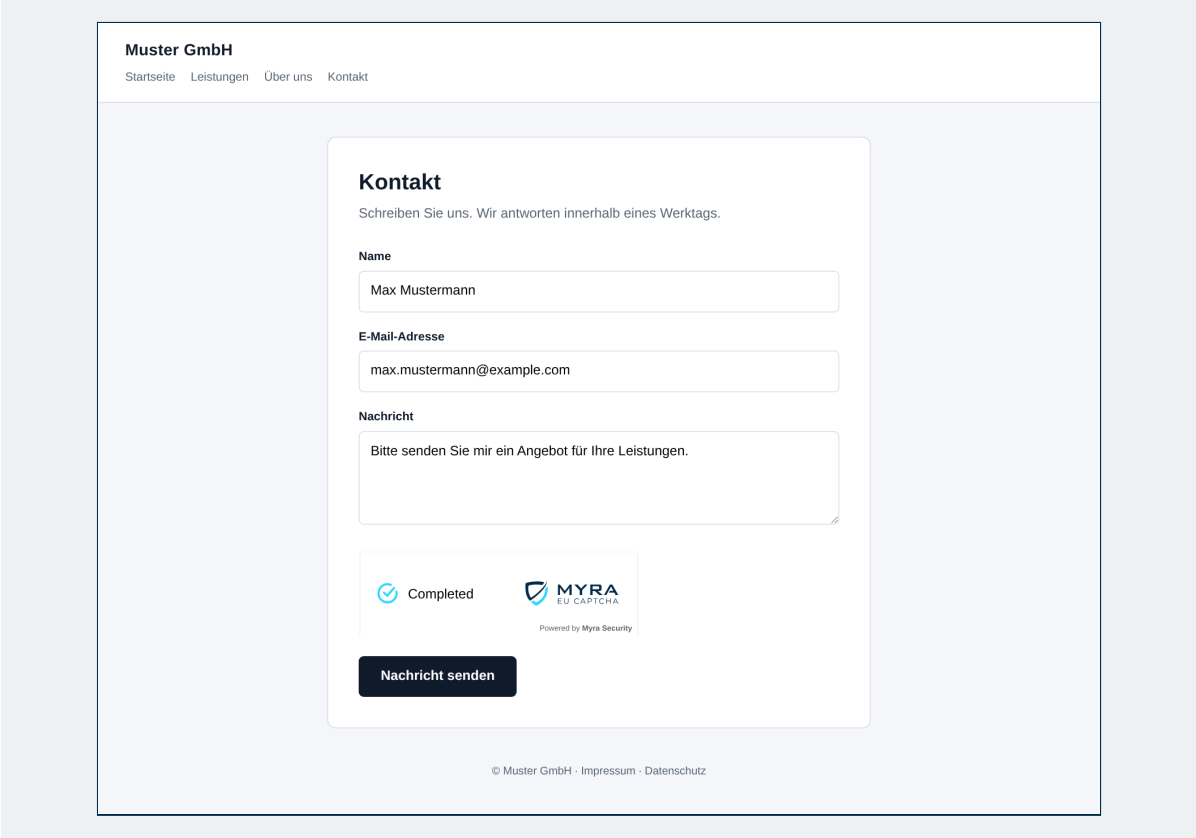


Abbildung 14: Geschütztes Formular auf der Website des Kunden

- ► Öffnen Sie das geschützte Formular.
- ► Senden Sie das Formular ab.
  - ↳ Das Widget löst die Challenge im Hintergrund, und das Formular wird verarbeitet.
- ► Öffnen Sie im Kundenportal die Ansicht **Challenges** des Sitekeys.
- ➔ Die Liste enthält einen Eintrag für die soeben gelöste Challenge.

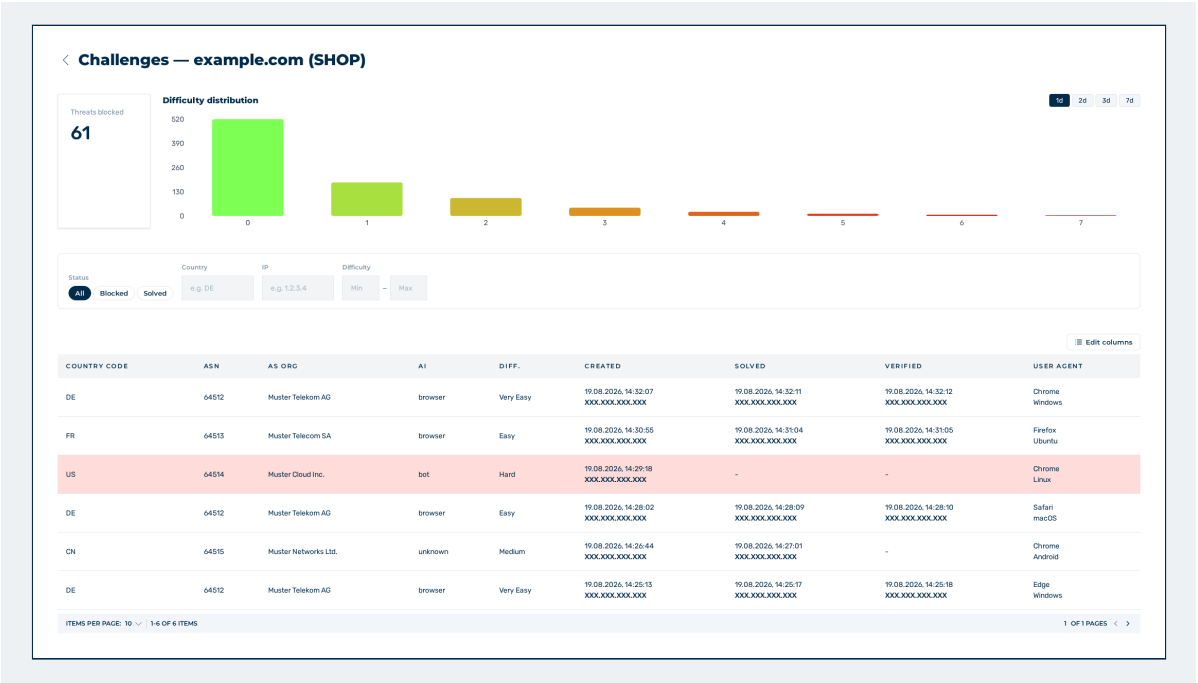


Abbildung 15: Ansicht Challenges

Siehe **Ansicht Challenges**.

2.2.3.2 Nach der Einrichtung

Der Sitekey ist einsatzbereit. Die folgenden Schritte sind sinnvoll, aber nicht zwingend:

Schritt	Beschreibung	Beschrieben in
Sitekey konfigurieren	Schwierigkeitsgrad, Zeitvorgabe, parallele Challenges und den Schalter **EU CAPTCHA Protection** einstellen.	**Sitekey konfigurieren**
Zugriff gewähren	Weiteren Konten lesenden oder schreibenden Zugriff auf den Sitekey geben.	**Zugriff auf einen Sitekey gewähren**
Verkehr beobachten	Die Challenges und ihre Ergebnisse je Sitekey auswerten.	**Ansicht Challenges**
Plan buchen	Vor dem Ende der Testphase einen Plan wählen.	**Testphase**

Solange der Schalter **EU CAPTCHA Protection** auf **Off** steht, stellt das Widget keine Challenge und die Anfragen laufen ungeprüft durch. Der Integrationstest kann dann gelingen, ohne dass ein Schutz besteht. Stellen Sie den Schalter vor dem Wirkbetrieb auf **On**.

2.3 Testphase

Jedes neue Konto beginnt mit einer kostenlosen Testphase von 30 Tagen und vollem Zugriff auf alle Funktionen. Für den Beginn der Testphase ist keine Zahlungsangabe nötig.

Nach der Testphase folgt eine Nachfrist von sieben Tagen. Danach ruht das Konto, bis ein Plan gewählt ist.

In den nächsten Abschnitten werden die folgenden Themen behandelt:

- **Plan während der Testphase festlegen** beschreibt, wie der Plan **Captcha Starter** ohne sofortige Abbuchung vorgemerkt wird.

2.3.1 Verlauf der Testphase

Das Kundenportal zeigt den Stand der Testphase in einem Banner am oberen Rand jeder Seite an. Das Banner wechselt mit der verbleibenden Zeit:

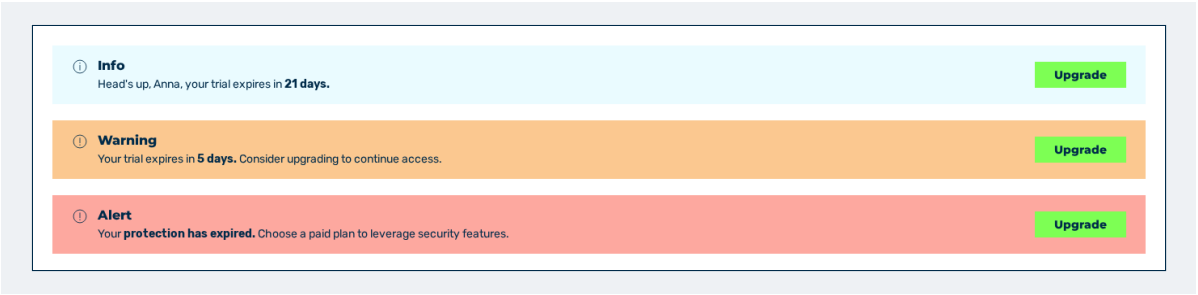


Abbildung 16: Die drei Banner der Testphase: Info, Warning und Alert

Die folgenden Zustände sind möglich:

Zustand	Banner	Meldung
Mehr als 7 Tage verbleiben	Blaues Banner **Info**	**Head's up, [Vorname], your trial expires in [N] days.**
7 Tage oder weniger verbleiben	Gelbes Banner **Warning**	**Your trial expires in [N] days. Consider upgrading to continue access.**
Nachfrist	Rotes Banner **Alert**	**Your protection has expired. Choose a paid plan to leverage security features.**
Nach der Nachfrist	Sperre über dem Kundenportal	**TRIAL ENDED**

Jedes Banner enthält die Schaltfläche **Upgrade**. Sie führt in den Reiter **Plans** der Ansicht **Account**. Siehe Reiter Plans.

2.3.2 Nachfrist

Die Nachfrist beginnt mit dem Ende der Testphase und dauert sieben Tage. Während der Nachfrist bleibt das Kundenportal erreichbar.

2.3.3 Nach der Nachfrist

Ist auch die Nachfrist abgelaufen, legt sich die Sperre **TRIAL ENDED** über das Kundenportal. Die Schaltfläche **Choose a plan** führt in den Reiter **Plans**.

Der Zustand hat die folgenden Folgen:

Bereich	Folge
Sitekeys	Die Sitekeys werden auf den Zustand **trial_expired** gesetzt. Eingebundene Widgets bleiben bestehen, blockieren aber keine Anfragen mehr. Der Schutz ist damit aufgehoben.
Secrets	Neue Secrets lassen sich nicht mehr anlegen. Der Dienst antwortet mit dem Fehler `TRIAL_EXPIRED` .
Konto	Das Konto bleibt bestehen und wird nicht gelöscht.

Um den Schutz wiederherzustellen, wählen Sie im Reiter **Plans** einen Plan. Siehe **Plan buchen** und **Testphase abgelaufen**.

Während einer laufenden Testphase lässt sich ein Plan im Voraus buchen, ohne dass sofort abgebucht wird. Der Plan trägt dann die Marke **Committed** und startet nach dem Ende der Testphase. Siehe **Plan während der Testphase festlegen**.

2.3.4 Plan während der Testphase festlegen

Während einer laufenden Testphase lässt sich der Plan **Captcha Starter** im Voraus festlegen, ohne dass sofort abgebucht wird. Das Kundenportal hinterlegt die Zahlungsart und bucht erst am Ende der Testphase ab. Der Plan wird zum selben Zeitpunkt automatisch aktiv.

Alle anderen Pläne sowie jeder Kauf nach dem Ende der Testphase laufen über den gewöhnlichen Bezahlvorgang und werden sofort abgebucht. Siehe **Plan buchen**.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Ihr Konto befindet sich in einer laufenden Testphase. Siehe **Testphase**.

☑ Für Ihr Konto besteht weder ein bezahlter Plan noch eine Vormerkung.

Um den Plan **Captcha Starter** festzulegen, gehen Sie wie folgt vor:

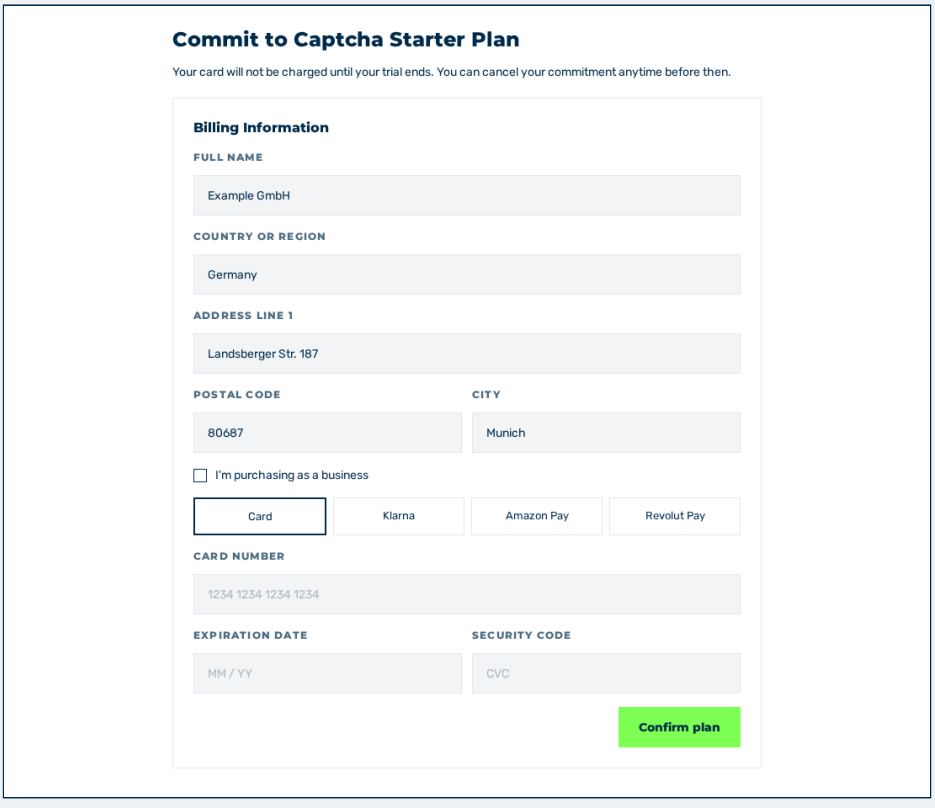


Abbildung 17: Ansicht Commit to Captcha Starter Plan mit dem Bereich Billing Information

- ▸ Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile** und wechseln Sie in den Reiter **Plans**.
- ▸ Klicken Sie im Plan **Starter** auf die Schaltfläche **Buy now**.

↳ Die Ansicht **Commit to Captcha Starter Plan** wird mit dem Hinweis **Your card will not be charged until your trial ends. You can cancel your commitment anytime before then.** angezeigt.

- ▸ Füllen Sie im Bereich **Billing Information** die Rechnungsanschrift aus.
- ▸ Wählen Sie eine Zahlungsart aus und geben Sie die zugehörigen Daten ein.
- ▸ Klicken Sie auf die Schaltfläche **Confirm plan**.

↳ Während der Verarbeitung trägt die Schaltfläche die Beschriftung **Processing...**.

→ Die Ansicht **You're all set!** wird angezeigt und nennt das Datum der ersten Abbuchung.

Die folgenden Zahlungsarten stehen zur Verfügung:

Zahlungsart	Beschreibung
Card	Kredit- oder Debitkarte.
Klarna	Zahlung über Klarna.
Amazon Pay	Zahlung über Amazon Pay.
Revolut Pay	Zahlung über Revolut Pay.

ⓘ Als Unternehmen aktivieren Sie das Kontrollkästchen **I'm purchasing as a business** und tragen Ihre Umsatzsteuer-Identifikationsnummer in das Feld **VAT Number (optional)** ein. Tragen Sie in das Feld **Full name** den Namen des Unternehmens ein. Dieser Name erscheint auf der Rechnung.

2.3.4.1 Vormerkung zurücknehmen

Die Vormerkung lässt sich bis zum Ende der Testphase jederzeit zurücknehmen. Bis dahin ist keine Zahlung erfolgt.

Ist ein Plan festgelegt, nennt die Marke über dem Angebot im Reiter **Plans** den Starttermin, zum Beispiel **Captcha Starter Plan committed — activates on 13.06.2026**. Der Reiter **Subscription History** führt den Eintrag mit der Marke **Committed** und der Schaltfläche **Cancel commitment**.

Wie eine Vormerkung zurückgenommen wird, steht unter **Abonnement kündigen**.

Die Testphase läuft nach der Rücknahme bis zu ihrem ursprünglichen Enddatum weiter. Der Dialog nennt dieses Datum.

3. Ansichten

3.1 Allgemein

3.1.1 Ansichten

Das Kundenportal des Myra EU CAPTCHA gliedert sich in eine Seitenleiste und den Arbeitsbereich. Die Seitenleiste führt zu den Ansichten, der Arbeitsbereich zeigt die gewählte Ansicht an.

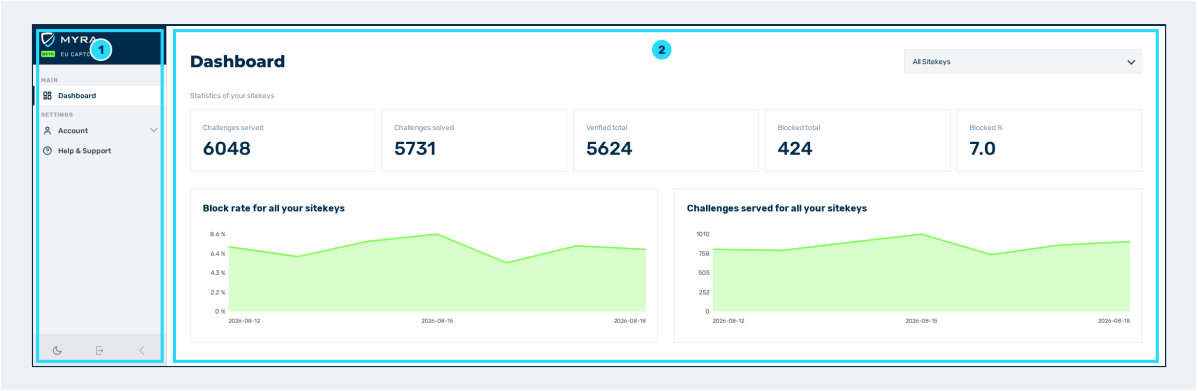


Abbildung 18: Aufbau des Kundenportals

3.1.1.1 Seitenleiste

Die Seitenleiste gliedert sich in Bereiche. Die folgenden Einträge stehen zur Verfügung:

Bereich	Eintrag	Ziel
MAIN	**Dashboard**	Ansicht **Dashboard** mit Kennzahlen, Diagrammen und der Liste der Sitekeys
SETTINGS	**Account** › **Profile**	Ansicht **Account** mit den Reitern zu Profil, Anmeldungen, Plänen, Abonnements und Löschung
–	**Help & Support**	Ansicht **Help** mit den Verweisen auf Dokumentation und Support

Der Eintrag **Account** ist aufklappbar. Ein Klick auf den Eintrag zeigt die untergeordneten Einträge an.

Am oberen Rand der Seitenleiste steht das Myra-Logo mit der Kennzeichnung **BETA**. Am unteren Rand stehen die folgenden Schaltflächen zur Verfügung:

- **Switch to dark mode** und **Switch to light mode** wechseln zwischen heller und dunkler Darstellung.
- **Logout** meldet Sie vom Kundenportal ab.
- **Collapse sidebar** und **Expand sidebar** verkleinern und vergrößern die Seitenleiste. Verkleinert zeigt die Seitenleiste nur die Symbole an. Der Eintragsname erscheint als Kurzhinweis.

> Der Bereich **ADMINISTRATION** mit dem Eintrag **Admin Control** ist der Administration durch Myra vorbehalten und in dieser Online-Hilfe nicht beschrieben.

3.1.1.2 Ansichten im Überblick

Die folgenden Ansichten stehen zur Verfügung:

Ansicht	Inhalt
Ansicht Dashboard	Kennzahlen und Diagramme zu den Challenges, darunter die Liste der Sitekeys
Bereich Sitekey-Liste	Tabelle aller Sitekeys mit Domain, Status und Optionen; Teil der Ansicht **Dashboard**
Ansicht Sitekey	Details, Konfiguration, Integration, Integrationstest, Berechtigungen und Challenges eines Sitekeys
Ansicht Account	Profil, Anmeldungen, Pläne, Abonnements und Löschung des Kontos
Ansicht Organization	Mitglieder der Organisation und Parameter für SAML und SSO
Ansicht Help	Verweise auf Dokumentation und Support

3.2 Dashboard

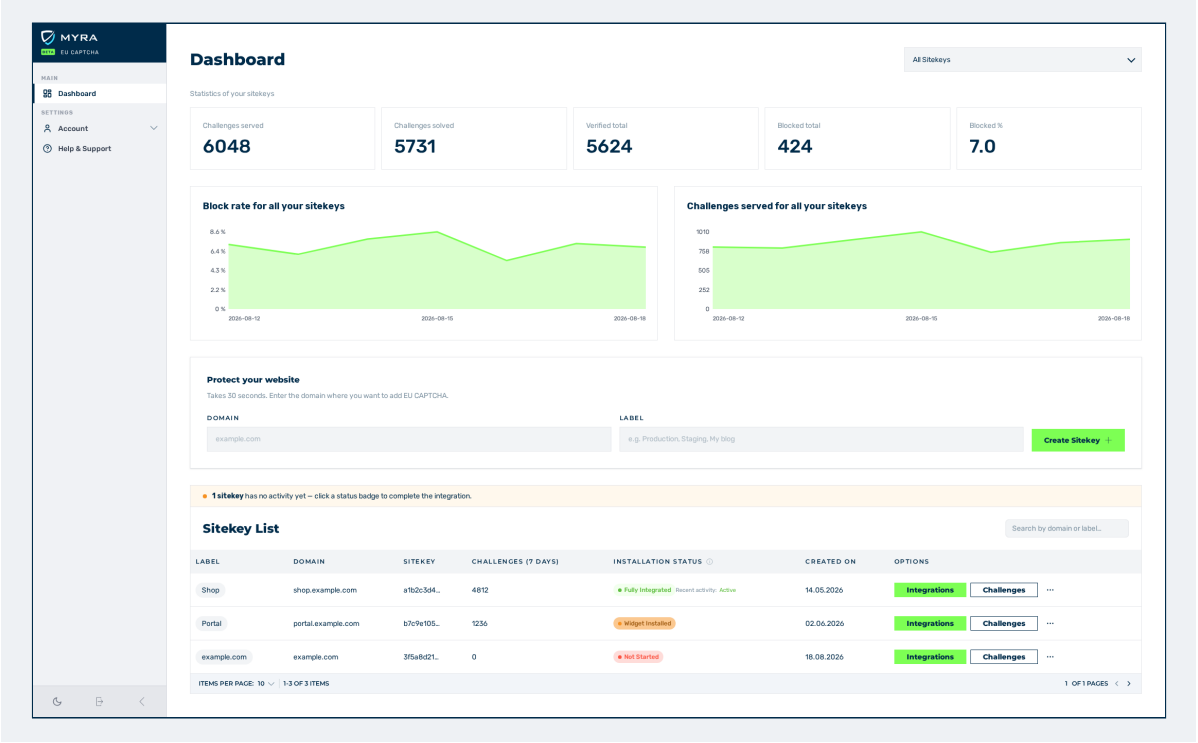


Abbildung 19: Ansicht Dashboard

Die Ansicht **Dashboard** ist die Startansicht des Kundenportals. Sie zeigt die Kennzahlen und den Verlauf der Challenges an. Darunter zeigt die Ansicht die Liste der Sitekeys an.

> **Video:** Die Ansicht Dashboard und ihre Kennzahlen — abspielbar in der Online-Hilfe

3.2.1 Auswahl der Sitekeys

Über der Kennzahlenreihe steht eine Drop-down-Liste mit den Sitekeys. Der Eintrag **All Sitekeys** fasst alle Sitekeys zusammen. Wenn Sie einen einzelnen Sitekey wählen, dann beziehen sich die Kennzahlen und die Diagramme ausschließlich auf diesen Sitekey.

Ein Suchfeld schränkt die Liste ein. Eine Schaltfläche setzt die Suche zurück.

3.2.2 Kennzahlen

Die Zeile **Statistics of your sitekeys** enthält die folgenden Kennzahlen:

Kennzahl	Bedeutung
Challenges served	Anzahl der ausgelieferten Challenges
Challenges solved	Anzahl der gelösten Challenges
Verified total	Anzahl der serverseitig bestätigten Token
Blocked total	Anzahl der blockierten Anfragen
Blocked %	Anteil der blockierten Anfragen an allen Challenges

3.2.3 Diagramme

Unter den Kennzahlen stehen zwei Verlaufsdiagramme:

Diagramm	Inhalt
Block rate for …	Verlauf der Blockierrate
Challenges served for …	Verlauf der ausgelieferten Challenges

Der Titel nennt die Bezeichnung des gewählten Sitekeys. Wenn Sie den Eintrag **All Sitekeys** gewählt haben, dann steht dort **all your sitekeys**.

Solange keine Daten vorliegen, zeigt das Diagramm den Hinweis **No data yet. Data appears within minutes of your first verified CAPTCHA challenge.** an.

3.2.4 Hinweise zum Einstieg

Je nach Stand der Einrichtung zeigt das Dashboard einen der folgenden Hinweise an:

Hinweis	Bedeutung	Schaltfläche
Protect your first website	Es ist noch kein Sitekey vorhanden.	**Add your first domain**
Complete your integration	Ein Sitekey ist vorhanden, es wurde jedoch noch keine Challenge ausgeliefert.	**Go to integration guide**

Nach dem Anlegen eines Sitekeys erscheint zusätzlich die Meldung **Sitekey created**.

3.2.5 Liste der Sitekeys

Unterhalb der Diagramme steht die Liste der Sitekeys. Siehe **Bereich Sitekey-Liste**.

3.2.6 Bereich Sitekey-Liste

Abbildung 20: Bereich Sitekey List

Der Bereich **Sitekey List** listet alle Sitekeys auf, auf die Sie Zugriff haben. Aus der Liste heraus öffnen Sie einen Sitekey, legen einen neuen an und löschen vorhandene Einträge.

> **Video:** Die Liste der Sitekeys und ihre Spalten — abspielbar in der Online-Hilfe

3.2.6.1 Spalten

Die Tabelle enthält die folgenden Spalten:

Spalte	Inhalt
Label	Bezeichnung des Sitekeys. Ohne Bezeichnung steht dort die Domain.
Domain	Geschützte Domain des Sitekeys
Sitekey	Erste acht Zeichen des öffentlichen Schlüssels
Challenges (7 days)	Anzahl der Challenges der letzten sieben Tage
Installation Status	Stand der Integration
Created on	Datum der Erstellung
Options	Aktionen für den Eintrag

Ein Suchfeld mit dem Platzhalter **Search by domain or label…** schränkt die Liste ein. Solange kein Sitekey vorhanden ist, zeigt die Tabelle den Hinweis **No Sitekeys added** an.

Über der Tabelle steht ein Hinweis, sobald mindestens ein Sitekey noch keine Challenge ausgeliefert hat: **N sitekeys have no activity yet — click a status badge to complete the integration.** Dabei steht `N` für die Anzahl der betroffenen Sitekeys.

Über dem neu angelegten Sitekey steht die Sprechblase **Click here for integration details**. Die Sprechblase verschwindet, sobald Sie die Schaltfläche **Integrations** anklicken.

Bei einem Sitekey mit abgeschaltetem Schutz steht in der Spalte **Label** zusätzlich die Marke **Training mode**.

Auf einen Sitekey, den ein anderes Konto geteilt hat, folgt in der Spalte **Label** die Marke **Read** oder **Write**.

3.2.6.2 Installationsstatus

Die Spalte **Installation Status** nimmt einen der folgenden Werte an:

Wert	Bedeutung
Not Started	Es wurde noch keine Aktivität des Widgets erkannt.
Widget Installed	Das Widget lädt, die serverseitige Prüfung fehlt noch.
Fully Integrated	Das Widget lädt, und Ihr Server prüft das Token.
Active	In den letzten 24 Stunden hat eine serverseitige Prüfung stattgefunden.
Inactive	In den letzten 24 Stunden hat keine serverseitige Prüfung stattgefunden.

Ein Kurzhinweis erläutert den jeweiligen Wert. Die Werte **Active** und **Inactive** stehen hinter der Bezeichnung **Recent activity:** und erscheinen erst beim Wert **Fully Integrated**.

3.2.6.3 Aktionen

Die Spalte **Options** enthält die folgenden Aktionen:

Aktion	Wirkung
Integrations	Öffnet den Sitekey mit der Ansicht **Details**.
Challenges	Öffnet die Ansicht **Challenges** des Sitekeys.
CONFIG	Öffnet den Dialog **Configuration Settings**.
PERMISSIONS	Öffnet die Ansicht **Permissions** des Sitekeys.
DELETE	Öffnet den Dialog **Delete Sitekey?** mit den Schaltflächen **Cancel** und **Delete**.

Die Aktionen **CONFIG**, **PERMISSIONS** und **DELETE** stehen im Menü hinter der Schaltfläche mit den drei Punkten. Die Aktionen **PERMISSIONS** und **DELETE** stehen nur beim Eigentümer des Sitekeys.

Siehe **Sitekey anlegen** und **Sitekey löschen**.

3.3 Sitekey

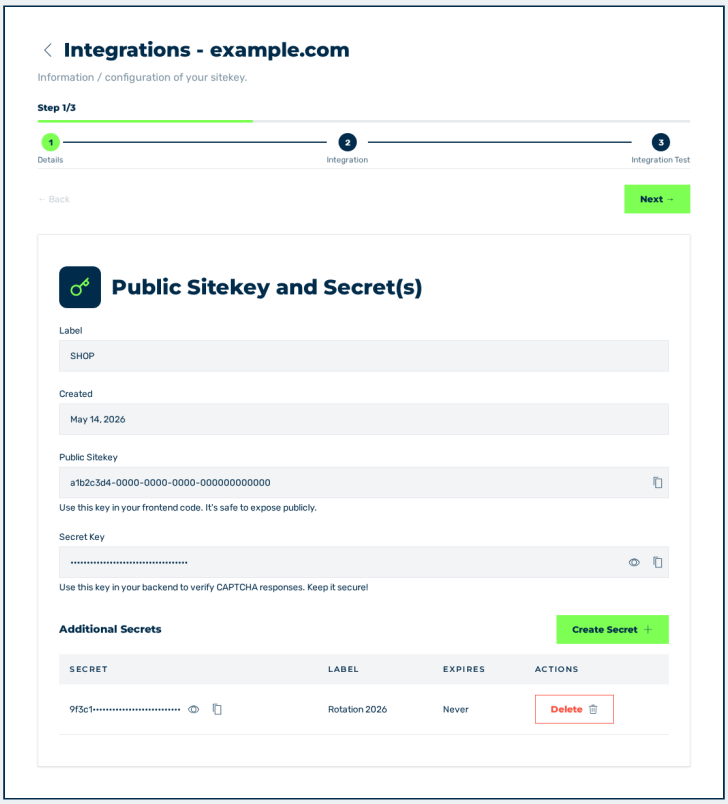


Abbildung 21: Ansicht Sitekey mit den Details des Sitekeys

Die Ansicht **Sitekey** fasst alle Angaben und Einstellungen eines Sitekeys zusammen. Sie öffnen die Ansicht aus der Liste der Sitekeys. Siehe **Bereich Sitekey-Liste**.

Über dem Inhalt steht die Überschrift **Sitekey -** , in den drei Schritten des Assistenten **Integrations -** . Darunter steht die Beschreibung **Information / configuration of your sitekey.**

3.3.1 Ansichten

Der Sitekey gliedert sich in die folgenden Ansichten:

Ansicht	Inhalt
Details	Öffentlicher Sitekey, Secrets und Stammdaten
Integration	Assistent mit der Anleitung für die gewählte Plattform
Integration Test	Prüfung von Widget und serverseitiger Verifizierung
Configuration	Schutz, Schwierigkeit, Zeitvorgabe und parallele Challenges
Permissions	Zugriff weiterer Konten auf den Sitekey
Challenges	Protokoll der ausgelieferten Challenges

3.3.2 Ablauf des Assistenten

Für die Ersteinrichtung führt ein Assistent durch drei der Ansichten:

Schritt	Ansicht
1	**Details**
2	**Integration**
3	**Integration Test**

Der Assistent zeigt über den Schritten die Angabe **Step ⅓** bis **Step 3/3** und einen Fortschrittsbalken an. Darunter stehen die drei Schritte mit ihrer Nummer und ihrer Bezeichnung.

Unter den Schritten und über dem Inhalt der Ansicht führt die Schaltfläche **Next** → zum nächsten Schritt. Die Schaltfläche ← **Back** führt zum vorherigen Schritt. Im ersten Schritt ist ← **Back** gesperrt, im letzten Schritt entfällt **Next** →.

Die Ansichten **Configuration** und **Permissions** gehören nicht zum Assistenten und zeigen die Schritte nicht an.

3.3.3 Zugriffsstufe

Ihre Zugriffsstufe auf den Sitekey bestimmt, ob Sie Einstellungen ändern dürfen. Mit der Stufe **Read** sind die Eingabefelder gesperrt. Ein Kurzhinweis nennt den Grund **Read-only access**. Siehe **Ansicht Permissions**.

3.3.4 Assistent

3.3.4.1 Details

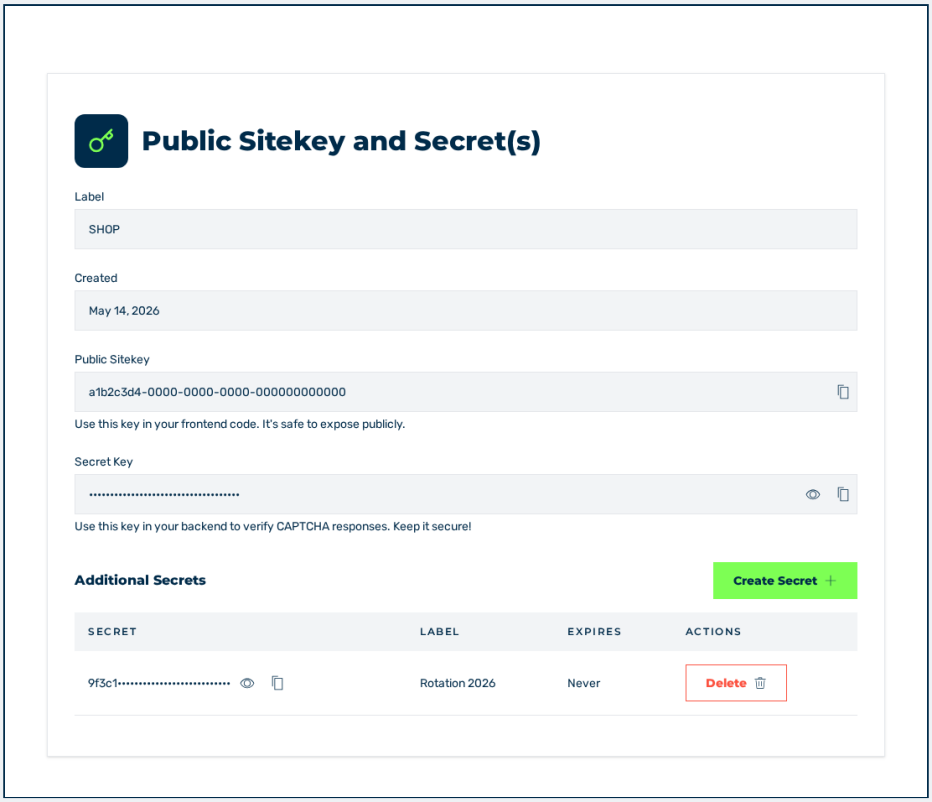


Abbildung 22: Ansicht Details

Die Ansicht **Details** zeigt unter der Überschrift **Public Sitekey and Secret(s)** die Schlüssel des Sitekeys und seine Stammdaten an.

3.3.4.1.1 Felder

Die Ansicht enthält die folgenden Felder:

Feld	Inhalt
Label	Bezeichnung des Sitekeys
Created	Datum der Erstellung
Public Sitekey	Öffentlicher Schlüssel für das Widget-Element
Secret Key	Geheimer Schlüssel für die serverseitige Prüfung

Der öffentliche Sitekey ist eine UUID. Er steht im Attribut `data-sitekey` des Widget-Elements und ist im Quelltext Ihrer Seite sichtbar.

Das Secret wird verdeckt dargestellt. Es gehört ausschließlich auf Ihren Server.

> Geben Sie das Secret niemals im HTML oder in JavaScript aus. Mit dem Secret lässt sich jedes Token als gültig ausgeben.

3.3.4.1.2 Schlüssel kopieren

Neben den Schlüsseln steht jeweils eine Schaltfläche zum Kopieren. Der Kurzhinweis lautet **Copy key to clipboard.** Nach dem Kopieren wechselt er auf **Key copied to clipboard.**

Neben dem Secret Key steht zusätzlich eine Schaltfläche, die den Wert einblendet. Der Kurzhinweis lautet **Show secret** und im eingeblendeten Zustand **Hide secret**.

3.3.4.1.3 Bereich Additional Secrets

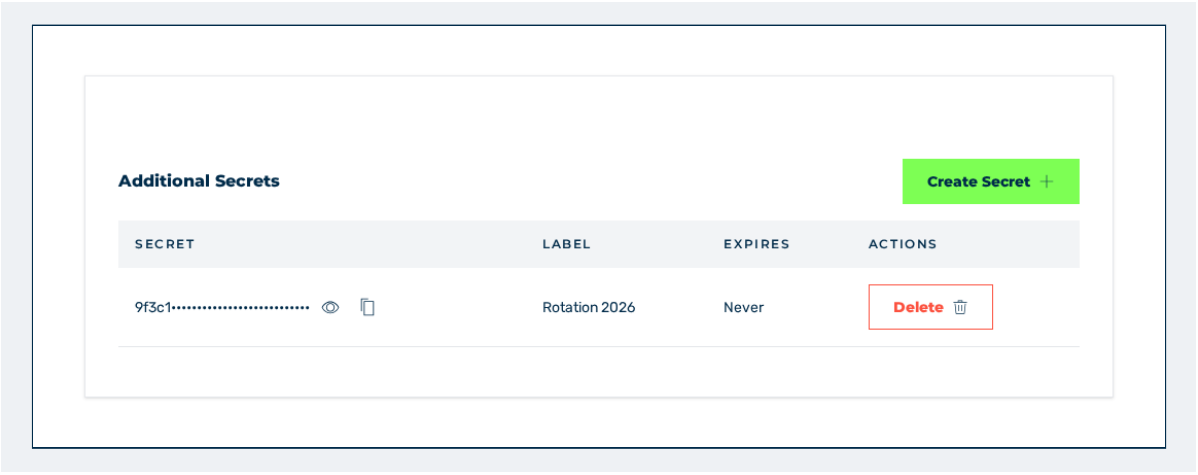


Abbildung 23: Bereich Additional Secrets

Der Bereich **Additional Secrets** verwaltet weitere Secrets zu diesem Sitekey. Mit der Zugriffsstufe **Read** zeigt die Ansicht den Bereich nicht an.

Weitere Secrets dienen dem Wechsel des Secrets im laufenden Betrieb: Sie legen ein zweites Secret an, stellen Ihre Server nacheinander darauf um und löschen anschließend das alte.

Die Tabelle enthält die folgenden Spalten:

Spalte	Inhalt
Secret	Secret, bis auf die ersten fünf Zeichen verdeckt
Label	Bezeichnung des Secrets, ohne Bezeichnung ein Bindestrich
Expires	Ablaufdatum, ohne Ablaufdatum der Wert **Never**
Actions	Schaltfläche **Delete**

Solange kein weiteres Secret vorhanden ist, zeigt die Tabelle den Hinweis **No additional secrets** an.

In der Spalte **Secret** blendet eine Schaltfläche den Wert ein und aus. Der Kurzhinweis lautet **Show secret** und **Hide secret**. Eine zweite Schaltfläche kopiert den Wert. Der Kurzhinweis lautet **Copy to clipboard** und nach dem Kopieren **Copied!**

3.3.4.1.4 Dialog Create Secret

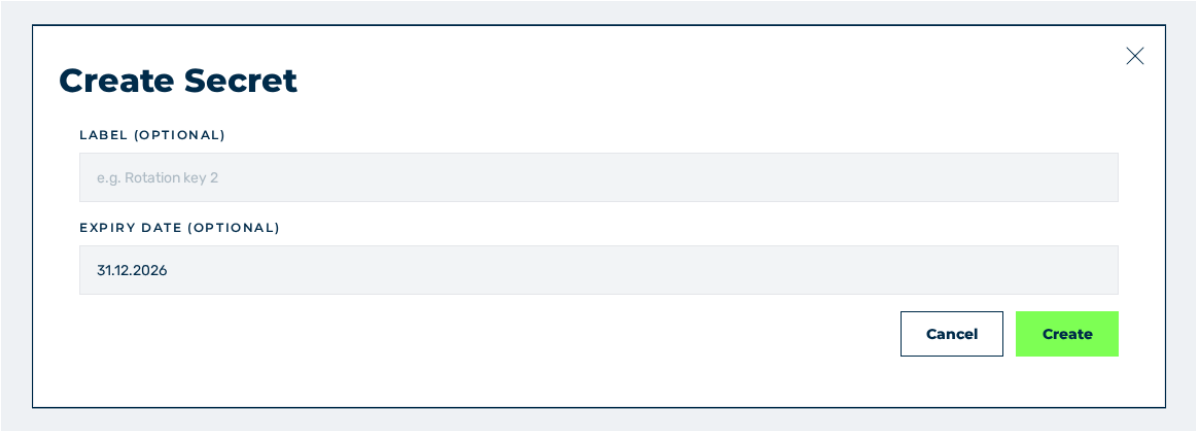


Abbildung 24: Dialog Create Secret

Die Schaltfläche **Create Secret** öffnet den Dialog **Create Secret**. Der Dialog enthält die folgenden Elemente:

Element	Beschreibung
Feld **Label (optional)**	Bezeichnung des Secrets, Platzhalter `e.g. Rotation key 2`
Feld **Expiry date (optional)**	Ablaufdatum des Secrets

Element	Beschreibung
Schaltfläche **Cancel**	Bricht den Vorgang ab
Schaltfläche **Create**	Legt das Secret an

3.3.4.1.5 Dialog Delete Secret

Abbildung 25: Dialog Delete Secret

Die Schaltfläche **Delete** öffnet den Dialog **Delete Secret** mit der Frage **Are you sure you want to delete the secret …? This action cannot be undone.** Die Schaltfläche **Delete** löscht das Secret, die Schaltfläche **Cancel** bricht den Vorgang ab.

Nach dem Löschen erscheint die Meldung **Secret deleted** mit dem Text **The secret has been removed**.

Siehe **Widget einbinden**.

3.3.4.2 Integration

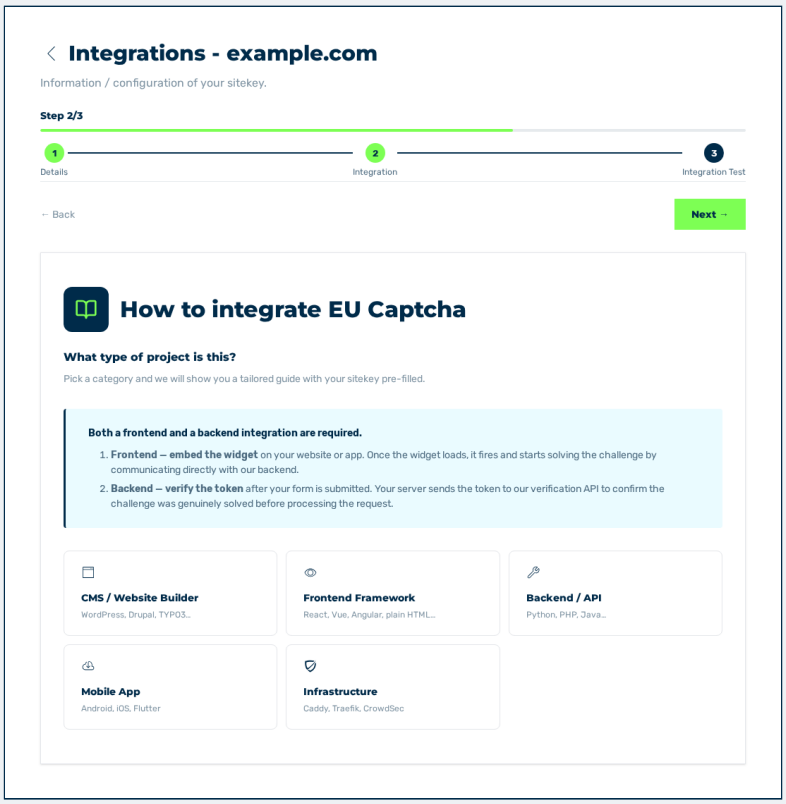


Abbildung 26: Ansicht Integration

Die Ansicht **Integration** führt unter der Überschrift **How to integrate EU Captcha** durch die Einbindung. Sie wählen die Plattform Ihrer Website, und der Assistent zeigt die passende Anleitung mit dem Sitekey dieses Eintrags an.

> **Video:** Die Ansicht Integration und die Auswahl der Plattform — abspielbar in der Online-Hilfe

3.3.4.2.1 Kategorien

Der Assistent bietet die folgenden Kategorien an:

Kategorie	Beispiele
CMS / Website Builder	WordPress, Drupal, TYPO3
Frontend Framework	React, Vue, Angular, reines HTML
Backend / API	Python, PHP, Java

Kategorie	Beispiele
Mobile App	Android, iOS, Flutter
Infrastructure	Caddy, Traefik, CrowdSec

Nach der Auswahl einer Kategorie zeigt der Assistent die Technologien dieser Kategorie an. Für Plattformen ohne eigene Bibliothek steht am Ende der Liste jeweils ein Eintrag für die Einbindung über HTML und die REST-API bereit.

3.3.4.2.2 Anleitung

Zur gewählten Technologie zeigt der Assistent die Schritte in aufklappbaren Abschnitten an. Die Anleitung enthält den öffentlichen Sitekey des geöffneten Eintrags, sodass Sie die Beispiele unverändert übernehmen können.

Je nach Technologie weist der Assistent zusätzlich darauf hin, dass der jeweils andere Teil der Integration noch fehlt:

- Bei Frontend-Technologien der Hinweis auf die serverseitige Prüfung.
- Bei Backend-Technologien der Hinweis auf die Einbindung des Widgets.

> ⓘ Ein Schutz entsteht erst, wenn beide Teile eingerichtet sind: das Widget in der Seite und die Prüfung des Tokens auf Ihrem Server.

Die ausführlichen Anleitungen dieser Online-Hilfe finden Sie unter **Integration**.

3.3.4.3 Integration Test

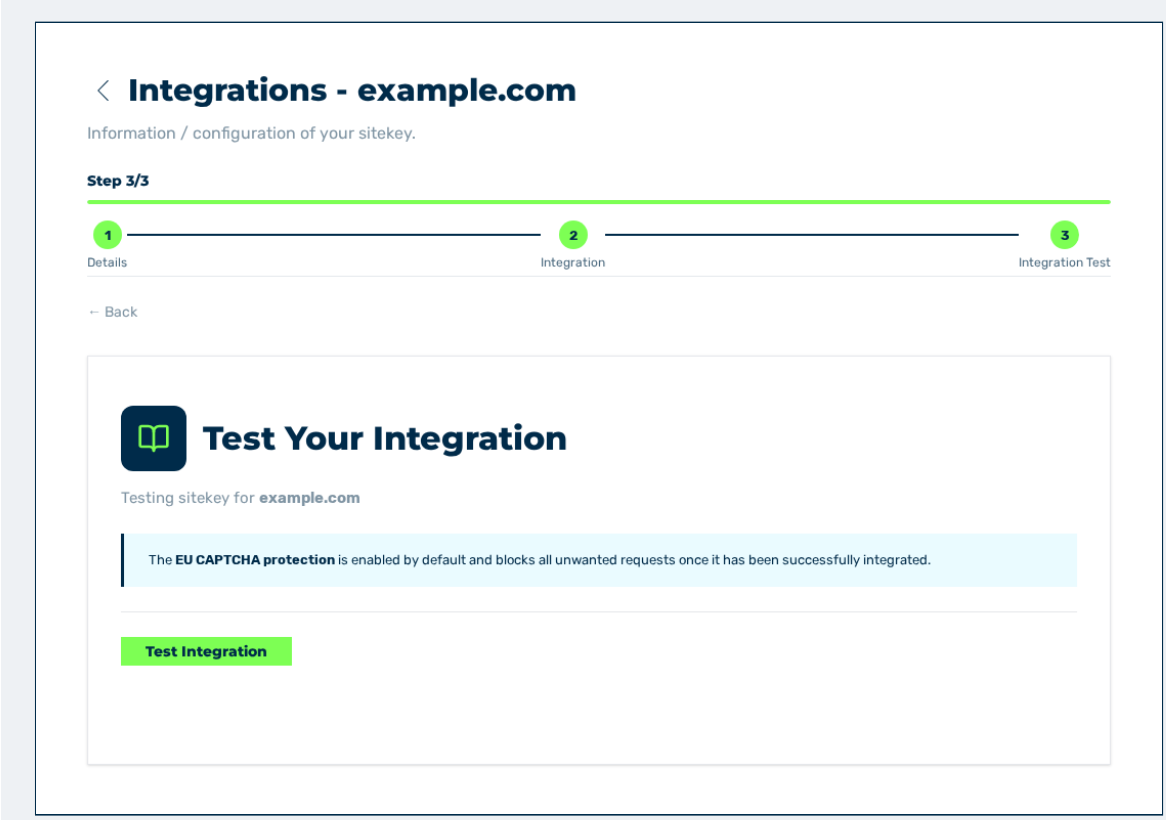


Abbildung 27: Ansicht Integration Test

Die Ansicht **Integration Test** prüft unter der Überschrift **Test Your Integration**, ob das Widget auf Ihrer Seite lädt und ob Ihr Server das Token prüft.

3.3.4.3.1 Ablauf

Die Schaltfläche **Test Integration** startet die Prüfung. Das Ergebnis nimmt einen der folgenden Werte an:

Ergebnis	Bedeutung
Fully Integrated	Das Widget lädt, und die serverseitige Prüfung hat stattgefunden.
Widget is working but your server didn't verify the token.	Das Widget lädt, die serverseitige Prüfung fehlt.
Widget not detected.	Das Widget wurde nicht erkannt.

Die Prüfung wertet die Aktivität des Sitekeys aus. Rufen Sie deshalb vor dem Test die geschützte Seite auf und senden Sie das Formular ab.

Siehe **Integration testen**.

3.3.5 Configuration

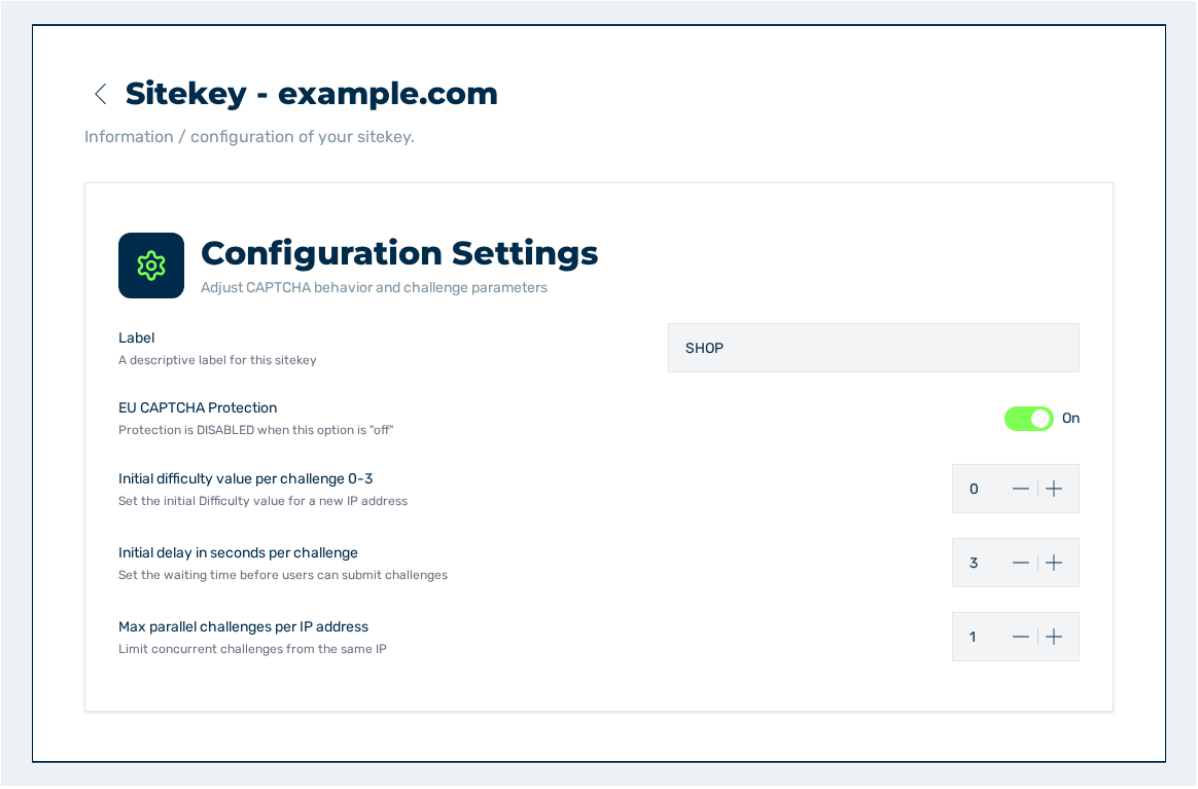


Abbildung 28: Ansicht Configuration

Die Ansicht **Configuration** steuert das Verhalten des Schutzes für diesen Sitekey. Die Überschrift lautet **Configuration Settings**, die Beschreibung **Adjust CAPTCHA behavior and challenge parameters**.

> **Video:** Die Ansicht Configuration und ihre Einstellungen — abspielbar in der Online-Hilfe

3.3.5.1 Einstellungen

Die Ansicht enthält die folgenden Einstellungen:

Einstellung	Beschreibung in der Oberfläche	Wertebereich	Vorgabe
Label	A descriptive label for this sitekey	Text	leer
EU CAPTCHA Protection	Protection is DISABLED when this option is off	**On**, **Off**	**On**
		0 bis 3	0

Einstellung	Beschreibung in der Oberfläche	Wertebereich	Vorgabe
Initial difficulty value per challenge 0-3	Set the initial Difficulty value for a new IP address		
Initial delay in seconds per challenge	Set the waiting time before users can submit challenges	3 bis 30	3
Max parallel challenges per IP address	Limit concurrent challenges from the same IP	1 bis 10	1

Wenn ein Wert außerhalb des Bereichs liegt, dann zeigt das Feld einen Hinweis auf die Ober- oder Untergrenze an.

3.3.5.2 Sperren

Zwei Bedingungen sperren die Einstellungen:

Bedingung	Kurzhinweis	Wirkung
Zugriffsstufe **Read**	**Read-only access**	Alle Einstellungen sind gesperrt.
Tarif **free**	**Upgrade to unlock**	Die drei Zahlenwerte sind gesperrt. Die Felder **Label** und **EU CAPTCHA Protection** bleiben änderbar. Die Schaltfläche **Upgrade** führt zur Auswahl der Pläne.

Siehe **Sitekey konfigurieren** und **Reiter Plans**.

3.3.6 Permissions

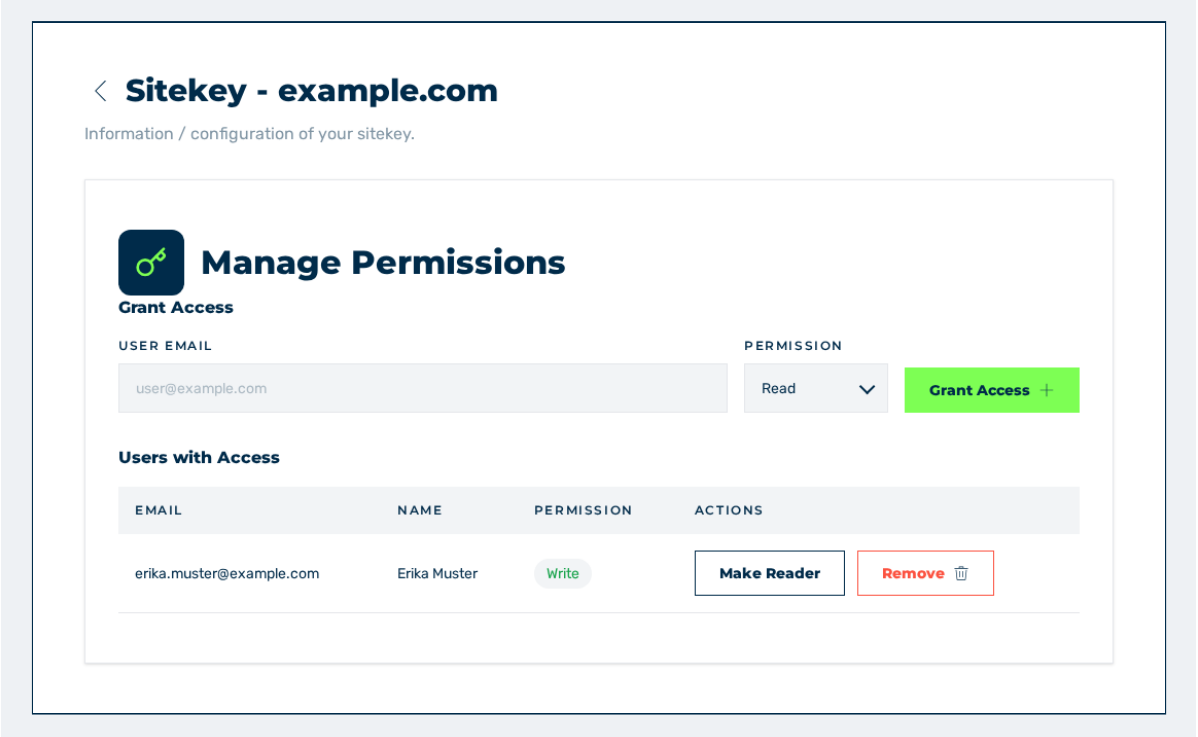


Abbildung 29: Ansicht Permissions

Die Ansicht **Permissions** steuert unter der Überschrift **Manage Permissions**, welche weiteren Konten auf diesen Sitekey zugreifen. Der Zugriff gilt jeweils für einen einzelnen Sitekey.

> **Video:** Die Ansicht Permissions und die Vergabe von Zugriffsrechten — abspielbar in der Online-Hilfe

3.3.6.1 Bereich Grant Access

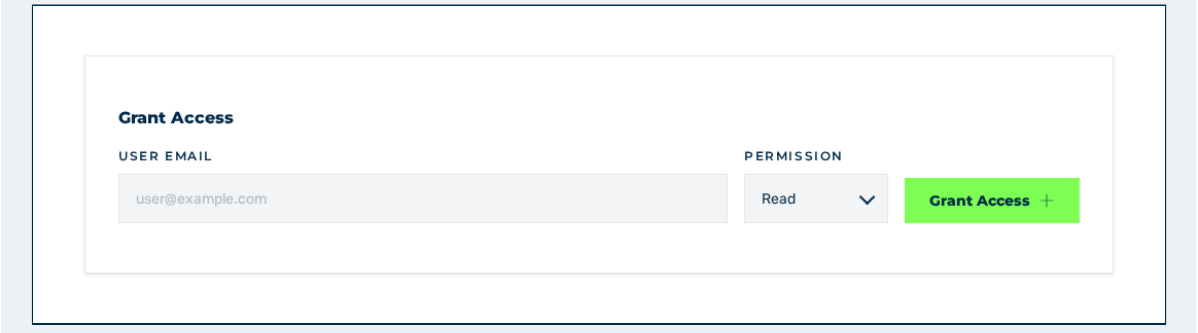


Abbildung 30: Bereich Grant Access

Der Bereich **Grant Access** vergibt den Zugriff. Er enthält die folgenden Elemente:

Element	Beschreibung
Feld **User Email**	E-Mail-Adresse des Kontos, Platzhalter `user@example.com`
Drop-down-Liste **Permission**	Zugriffsstufe **Read** oder **Write**
Schaltfläche **Grant Access**	Vergibt den Zugriff

3.3.6.2 Bereich Users with Access

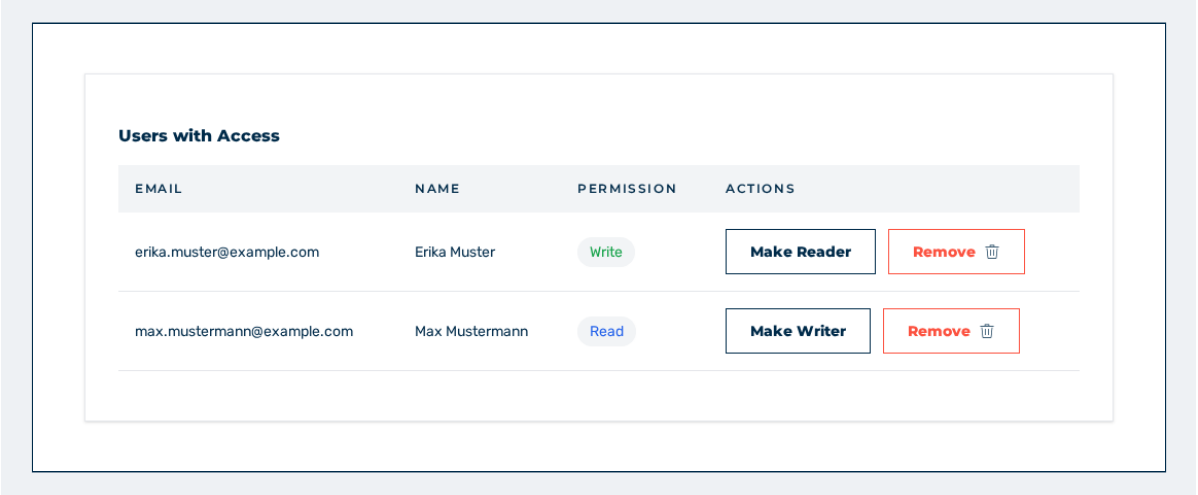


Abbildung 31: Bereich Users with Access

Der Bereich **Users with Access** listet die berechtigten Konten auf. Die Tabelle enthält die folgenden Spalten:

Spalte	Inhalt
Email	E-Mail-Adresse des Kontos
Name	Name des Kontos
Permission	Zugriffsstufe **Read** oder **Write**
Actions	Schaltflächen **Make Writer** und **Make Reader** sowie **Remove**

Solange kein weiteres Konto berechtigt ist, zeigt die Tabelle den Hinweis **No users have been granted access** an.

3.3.6.3 Zugriffsstufen

Die folgenden Zugriffsstufen stehen zur Verfügung:

Stufe	Rechte
Read	Der Sitekey und seine Einstellungen sind sichtbar. Änderungen sind gesperrt.
Write	Der Sitekey und seine Einstellungen sind sichtbar und änderbar.

3.3.6.4 Dialoge

Die Aktionen öffnen jeweils einen Dialog zur Bestätigung:

Dialog	Frage
Remove Permission	**Are you sure you want to remove access for …?**
Change to Write und **Change to Read**	**Change permission for … from … to …?**

Beide Dialoge enthalten die Schaltflächen **Cancel** und **Confirm**.

Siehe **Zugriff auf einen Sitekey gewähren**.

3.3.7 Challenges

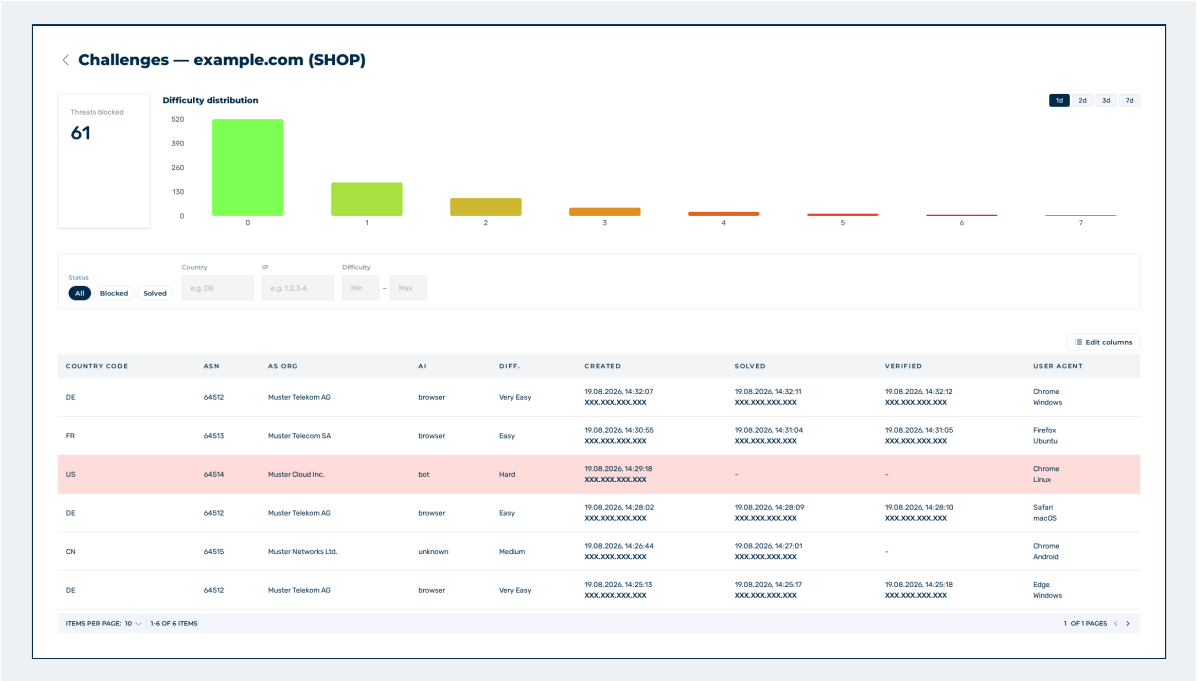


Abbildung 32: Ansicht Challenges

Die Ansicht **Challenges** protokolliert die Challenges dieses Sitekeys. Aus dem Protokoll erkennen Sie, welche Anfragen blockiert und welche freigegeben wurden. Die Überschrift lautet **Challenges – ()**.

> **Video:** Das Protokoll der Challenges und seine Filter — abspielbar in der Online-Hilfe

3.3.7.1 Spalten

Die Tabelle enthält die folgenden Spalten:

Spalte	Inhalt
Country Code	Ländercode der Herkunft
ASN	Autonomes System der IP-Adresse
As ORG	Betreiber des autonomen Systems
AI	Einstufung der Anfrage: **browser**, **bot** oder **unknown**
Diff.	Schwierigkeitsgrad der Challenge
Created	Zeitpunkt und IP-Adresse der Challenge

Spalte	Inhalt
Solved	Zeitpunkt und IP-Adresse der Lösung
Verified	Zeitpunkt und IP-Adresse der serverseitigen Prüfung
User Agent	Browser und Betriebssystem der Anfrage

Eine blockierte Challenge steht auf einem farbig hinterlegten Zeilenhintergrund. Ohne Wert steht in der Zelle ein Bindestrich.

Solange keine Challenge vorliegt, zeigt die Tabelle den Hinweis **No challenges yet** an.

Die Schaltfläche **Edit columns** über der Tabelle öffnet eine Liste mit einem Kontrollkästchen je Spalte. Damit blenden Sie einzelne Spalten aus und wieder ein.

3.3.7.2 Schwierigkeitsgrade

Die Spalte **Diff.** zeigt den Schwierigkeitsgrad als Text an:

Wert	Anzeige
0	**Very Easy**
1 und 2	**Easy**
3 und 4	**Medium**
5 und 6	**Hard**
7	**Very Hard**

3.3.7.3 Filter

Über der Tabelle stehen die folgenden Filter zur Verfügung:

Filter	Bedienung
Status	Schaltflächen **All**, **Blocked** und **Solved**
Country	Feld für den Ländercode, Platzhalter `e.g. DE`
IP	Feld für die IP-Adresse, Platzhalter `e.g. 1.2.3.4`
Difficulty	Zwei Felder **Min** und **Max** für den Bereich 0 bis 7

Sobald ein Filter gesetzt ist, erscheint die Schaltfläche **Clear filters**. Sie setzt alle Filter zurück.

Die Filter wirken auf die geladene Seite der Tabelle, nicht auf den gesamten Bestand.

3.3.7.4 Verteilung der Schwierigkeit

Über den Filtern steht das Diagramm **Difficulty distribution**. Es zeigt als Balken, wie viele Challenges auf die Schwierigkeitsgrade 0 bis 7 entfallen. Die Schaltflächen **1d**, **2d**, **3d** und **7d** wählen den Zeitraum.

Links davon nennt die Kennzahl **Threats blocked** die Zahl der blockierten Anfragen im gewählten Zeitraum.

Siehe **Funktionsweise**.

3.4 Account

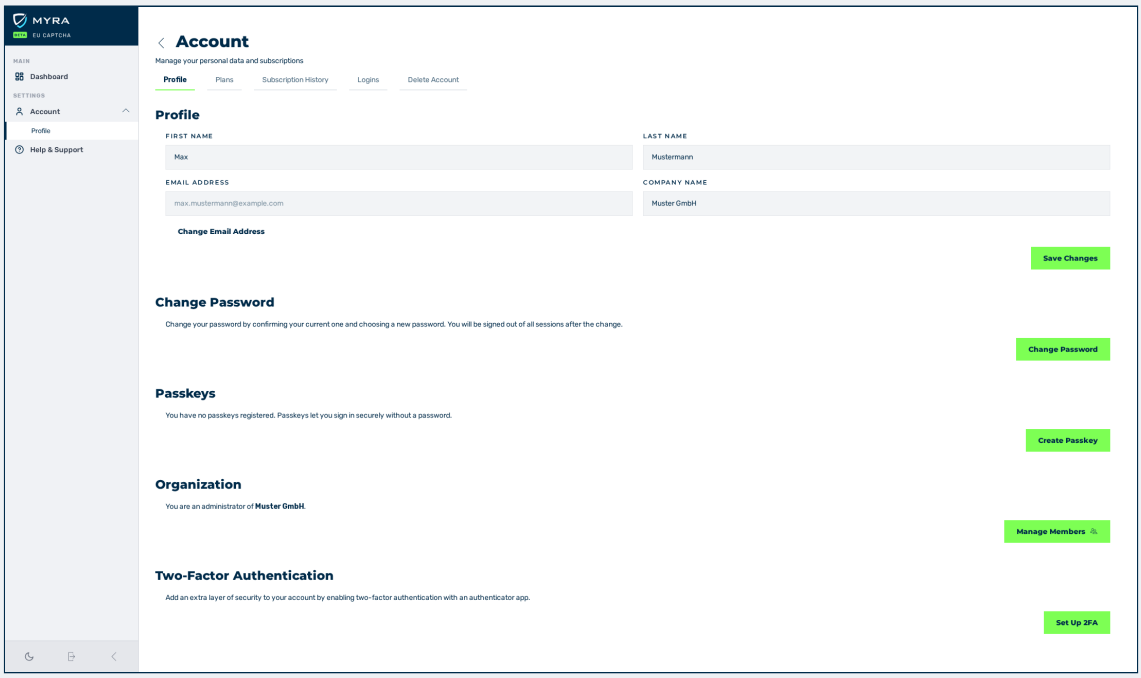


Abbildung 33: Ansicht Account

Die Ansicht **Account** fasst Ihre persönlichen Daten und Ihr Abonnement zusammen. Die Beschreibung unter dem Titel lautet **Manage your personal data and subscriptions**. Sie öffnen die Ansicht über den Eintrag **Account** › **Profile** der Seitenleiste.

> **Video:** Die Ansicht Account und ihre Reiter — abspielbar in der Online-Hilfe

3.4.1 Reiter

Die Ansicht gliedert sich in die folgenden Reiter:

Reiter	Inhalt
Profile	Persönliche Daten, Passwort, E-Mail-Adresse, Passkeys und Zwei-Faktor-Authentifizierung
Plans	Verfügbare Pläne und deren Buchung
Subscription History	Verlauf der Abonnements und deren Status
Logins	Protokoll der Anmeldungen an Ihrem Konto
Delete Account	Endgültige Löschung des Kontos

3.4.2 Reiter Profile

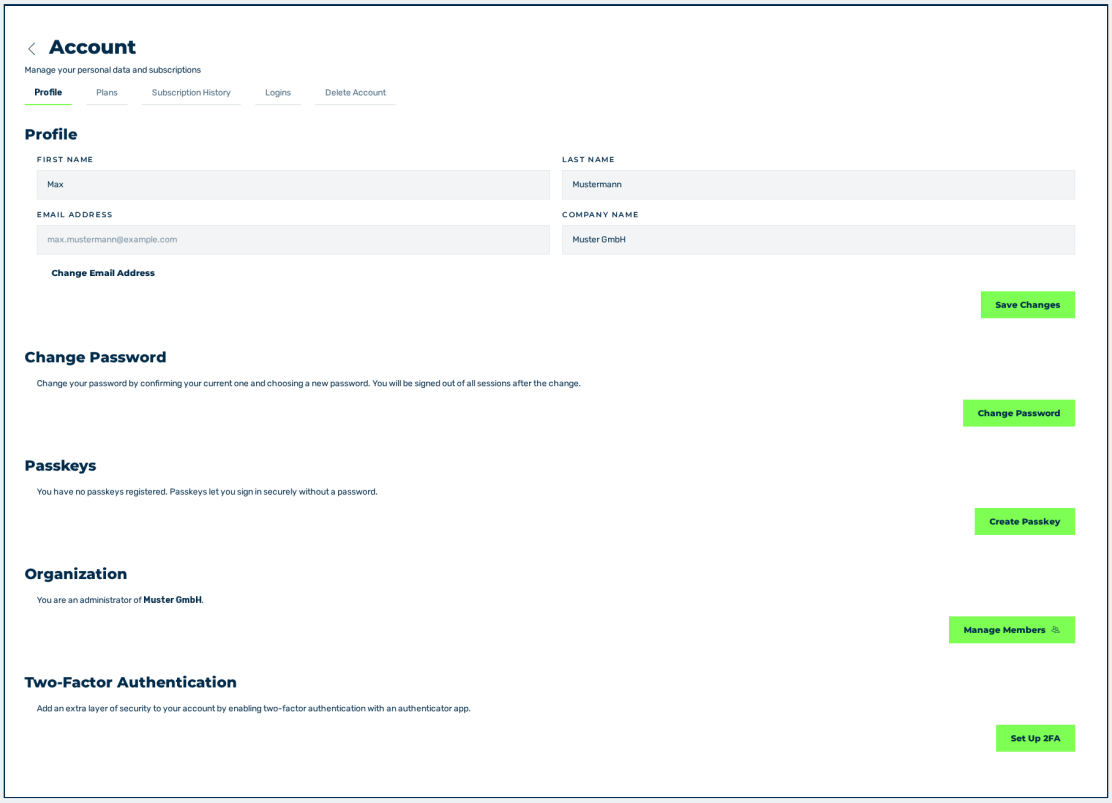


Abbildung 34: Reiter Profile

Der Reiter **Profile** enthält Ihre persönlichen Daten und die Einstellungen zur Sicherheit Ihres Kontos.

> **Video:** Der Reiter Profile und seine Einstellungen — abspielbar in der Online-Hilfe

3.4.2.1 Bereich Profile

Die folgenden Felder stehen zur Verfügung:

Feld	Inhalt
First name	Vorname
Last name	Nachname
Email address	E-Mail-Adresse des Kontos. Das Feld ist gesperrt.
Company name	Name des Unternehmens

Die Schaltfläche **Save Changes** übernimmt die Änderungen. Die Schaltfläche **Change Email Address** öffnet den gleichnamigen Dialog.

3.4.2.2 Bereich Change Password

Der Bereich beschreibt die Änderung des Passworts. Die Schaltfläche **Change Password** öffnet den gleichnamigen Dialog. Nach der Änderung meldet das System Sie aus allen Sitzungen ab.

3.4.2.3 Bereich Passkeys

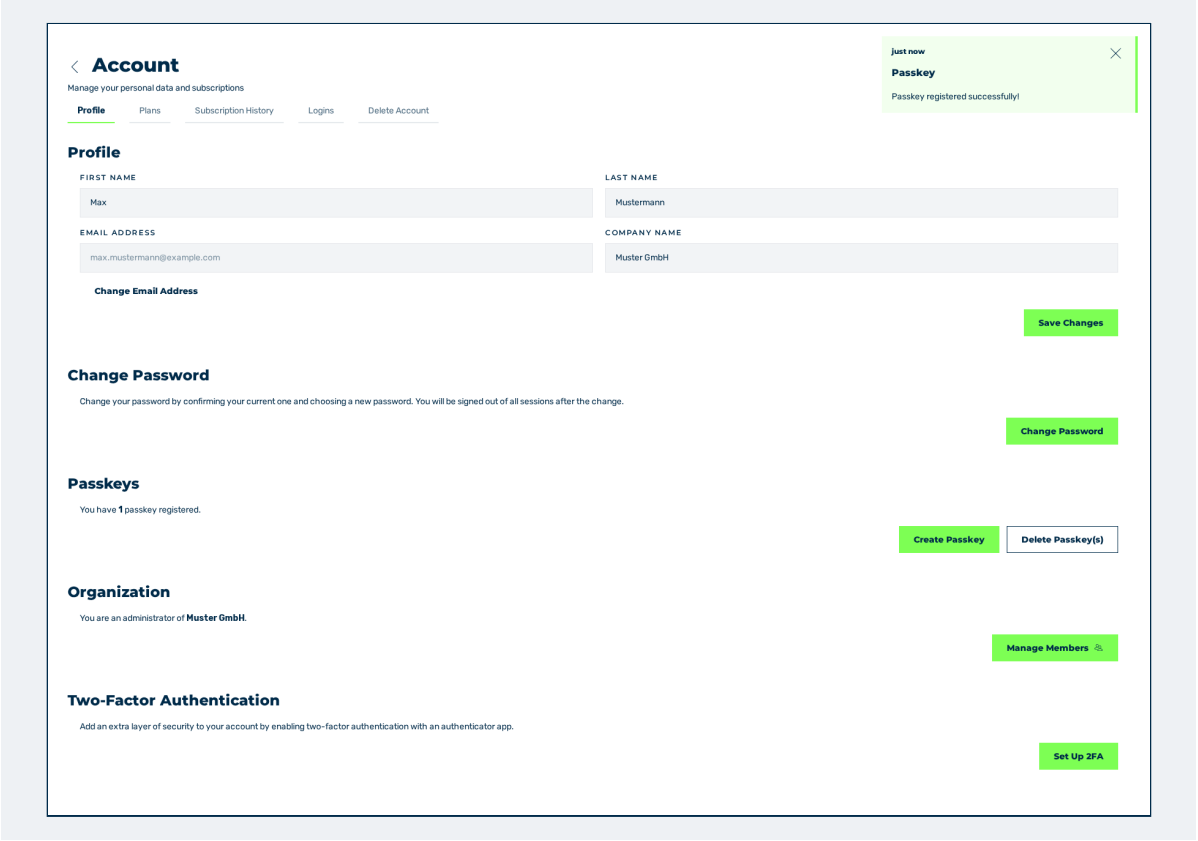


Abbildung 35: Reiter Profile mit einem angelegten Passkey

Der Bereich nennt die Anzahl der angelegten Passkeys. Die folgenden Schaltflächen stehen zur Verfügung:

Schaltfläche	Wirkung
Create Passkey	Legt einen Passkey für die Anmeldung ohne Passwort an.
Delete Passkey(s)	Löscht die vorhandenen Passkeys. Die Schaltfläche erscheint erst mit dem ersten Passkey.

3.4.2.4 Bereich Organization

Der Bereich erscheint nur, wenn Sie Org Admin einer Organisation sind. Er nennt den Namen der Organisation. Die Schaltfläche **Manage Members** öffnet die Verwaltung der Mitglieder.

3.4.2.5 Bereich Two-Factor Authentication

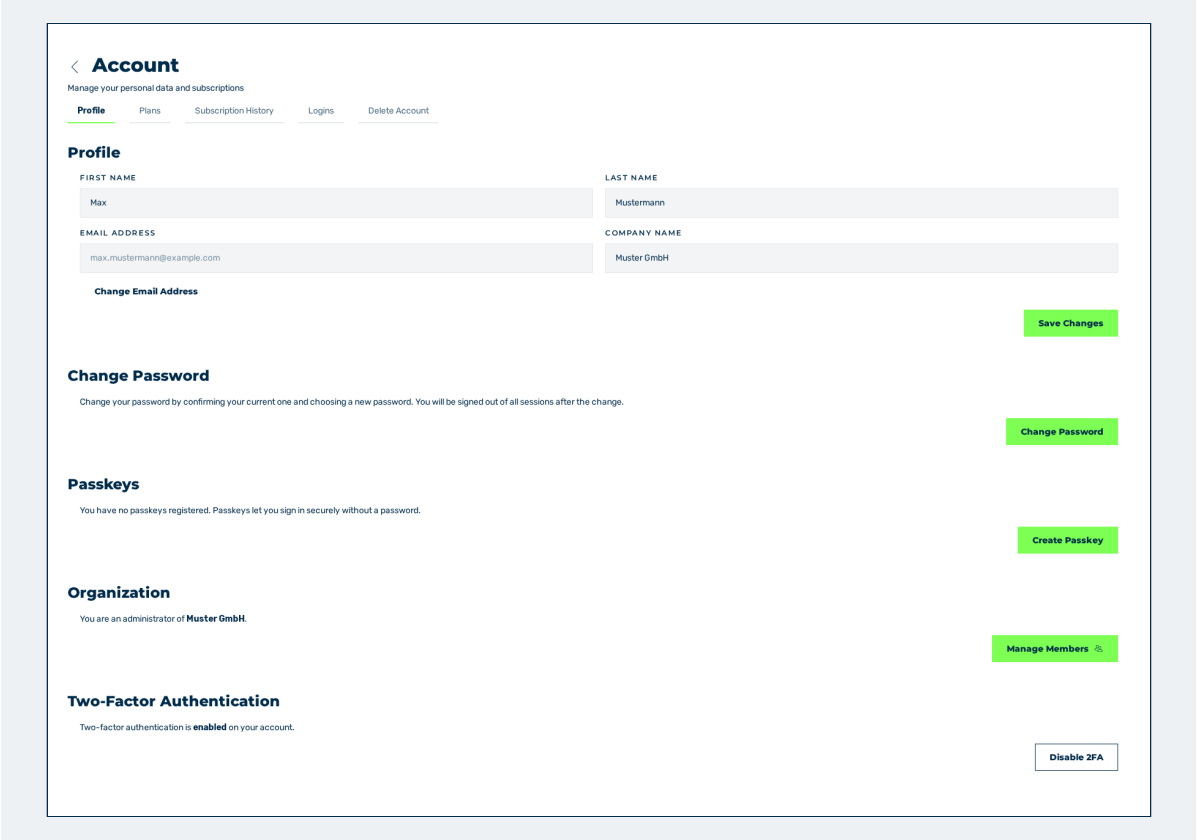


Abbildung 36: Reiter Profile mit aktivierter Zwei-Faktor-Authentifizierung

Der Bereich nennt den Zustand der Zwei-Faktor-Authentifizierung. Die folgenden Schaltflächen stehen zur Verfügung:

Schaltfläche	Wirkung
Set Up 2FA	Startet die Einrichtung der Zwei-Faktor-Authentifizierung.
Disable 2FA	Öffnet den Dialog **Disable Two-Factor Authentication**.

3.4.2.6 Meldungen

Das Portal bestätigt die Aktionen mit den folgenden Meldungen:

Meldung	Anlass
Password changed — Your password was changed. Please sign in again.	Das Passwort wurde geändert.
Email change requested — Check your current inbox to confirm the change.	Die Änderung der E-Mail-Adresse wurde angefordert.
Passkey — Passkey registered successfully!	Ein Passkey wurde angelegt.
Passkey — All passkeys have been deleted.	Alle Passkeys wurden gelöscht.
2FA — Two-factor authentication has been disabled.	Die Zwei-Faktor-Authentifizierung wurde deaktiviert.

Siehe **Passwort ändern**.

Siehe **E-Mail-Adresse ändern**.

Siehe **Zwei-Faktor-Authentifizierung einrichten**.

3.4.3 Reiter Plans

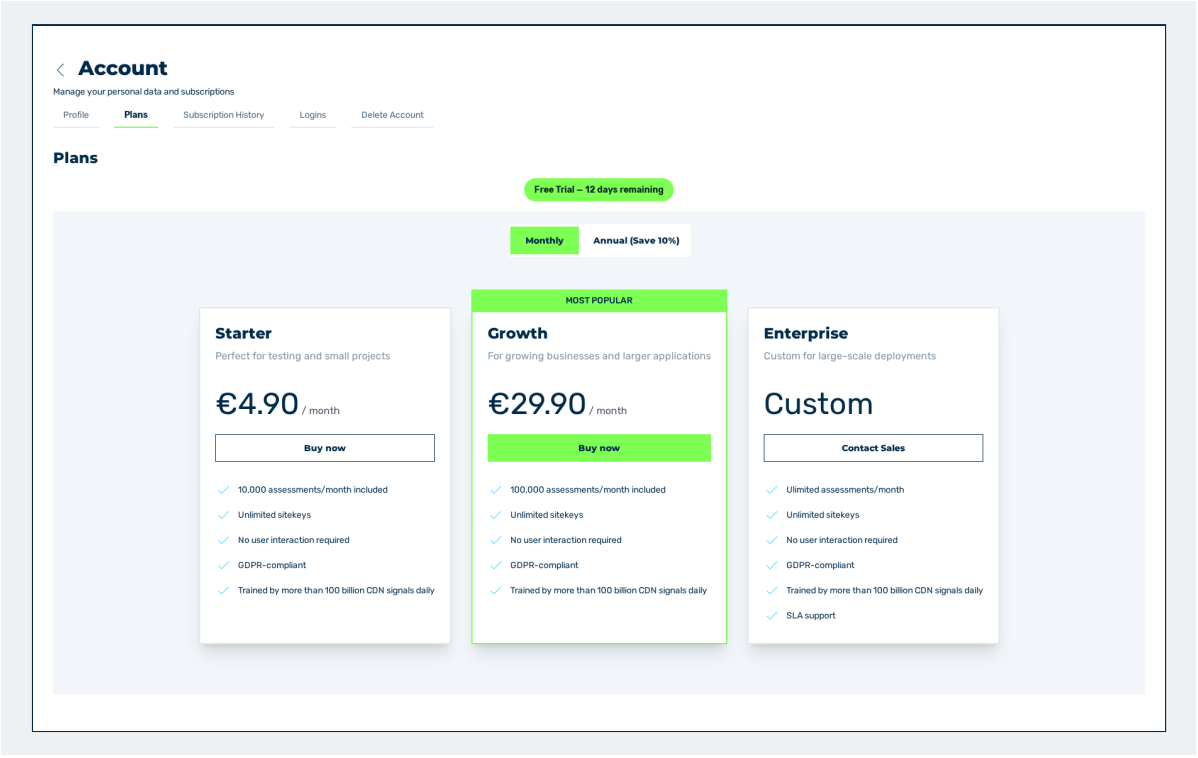


Abbildung 37: Reiter Plans

Der Reiter **Plans** zeigt die verfügbaren Pläne an und startet deren Buchung.

> **Video:** Die verfügbaren Pläne und ihre Buchung — abspielbar in der Online-Hilfe

3.4.3.1 Zustand des Plans

Über dem Angebot nennt eine farbige Marke den Zustand Ihres Plans:

Marke	Bedeutung
Free Trial – 12 days remaining	Die Testphase läuft. Die Marke nennt die verbleibenden Tage.
Free Trial Expired – 3 days to upgrade	Die Testphase ist abgelaufen. Die Marke nennt die verbleibenden Tage bis zur Sperre.
No active plan	Die Testphase ist abgelaufen, und es besteht kein Abonnement.
Captcha Starter Plan – Active	

Marke	Bedeutung
	Das Abonnement läuft. Die Marke nennt den gebuchten Plan.
Captcha Starter Plan committed – activates on 13.06.2026	Sie haben während der Testphase einen Plan gebucht. Die Marke nennt den Starttermin.
Plan cancelled – access until 19.09.2026	Sie haben das Abonnement gekündigt. Die Marke nennt das Ende des bezahlten Zeitraums.

3.4.3.2 Abrechnungszeitraum

Zwei Schaltflächen wählen den Abrechnungszeitraum:

Schaltfläche	Bedeutung
Monthly	monatliche Abrechnung
Annual (Save 10%)	jährliche Abrechnung mit einem Rabatt von 10 % auf den Monatspreis

3.4.3.3 Pläne

Die folgenden Pläne stehen zur Verfügung:

Plan	Beschreibung	Preis je Monat	Assessments je Monat	Schaltfläche
Starter	Perfect for testing and small projects	4,90 €	10.000	**Buy now**
Growth	For growing businesses and larger applications	29,90 €	100.000	**Buy now**
Enterprise	Custom for large-scale deployments	auf Anfrage	unbegrenzt	**Contact Sales**

Der Plan **Growth** trägt das Band **MOST POPULAR**.

Alle Pläne enthalten die folgenden Leistungen:

- unbegrenzt viele Sitekeys
- Prüfung ohne Zutun des Besuchers
- Konformität mit der DSGVO
- Bewertung anhand von täglich mehr als 100 Milliarden CDN-Signalen

Der Plan **Enterprise** enthält zusätzlich den Support nach SLA.

Bei jährlicher Abrechnung sinkt der Monatspreis auf 4,40 € für den Plan **Starter** und auf 26,90 € für den Plan **Growth**.

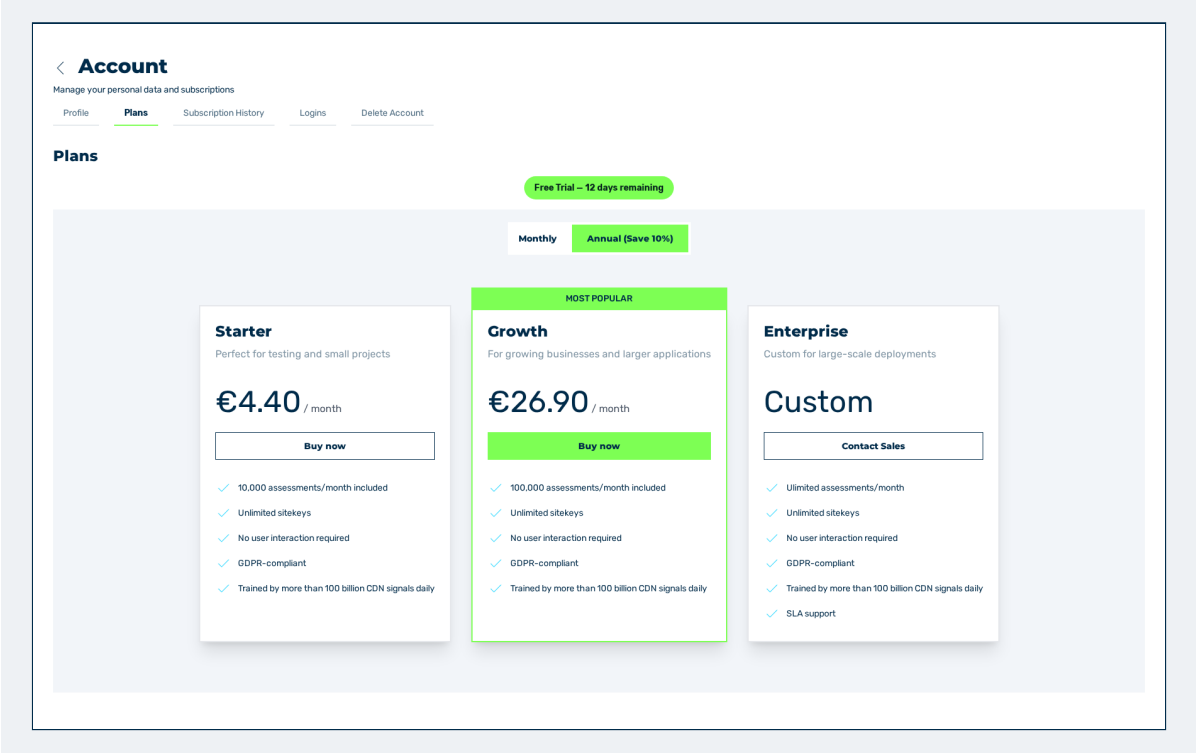


Abbildung 38: Reiter Plans mit jährlicher Abrechnung

Solange ein Abonnement läuft, tragen die Schaltflächen der Pläne **Starter** und **Growth** die Beschriftung **Already subscribed** und sind gesperrt.

> Die monatliche Obergrenze für Assessments wird derzeit nicht durchgesetzt. Ein Überschreiten der Grenze setzt das Widget nicht aus.

Siehe **Plan buchen**.

3.4.4 Reiter Subscription History

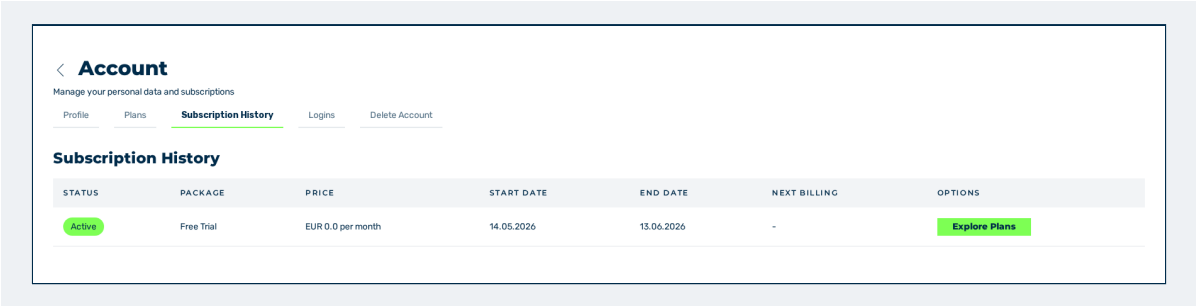


Abbildung 39: Reiter Subscription History

Der Reiter **Subscription History** zeigt den Verlauf Ihrer Abonnements an. Die Testphase steht als eigene Zeile mit dem Plan **Free Trial** in der Tabelle.

> **Video:** Der Verlauf der Abonnements — abspielbar in der Online-Hilfe

3.4.4.1 Spalten

Die Tabelle enthält die folgenden Spalten:

Spalte	Inhalt
Status	Zustand des Abonnements als farbige Marke
Package	Name des Plans: **Free Trial**, **Captcha Starter** oder **Captcha Growth**
Price	Betrag mit Währung und Zeitraum, zum Beispiel `EUR 4.90 per month`. Ein Jahresplan trägt den Zusatz `per year`.
Start date	Beginn des Abonnements im Format TT.MM.JJJJ
End date	Ende des laufenden Zeitraums im Format TT.MM.JJJJ
Next billing	Zeitpunkt der nächsten Abrechnung
Options	Schaltflächen für den Eintrag

3.4.4.2 Status

Die Spalte **Status** kennt die folgenden Marken:

Marke	Bedeutung
Active	Das Abonnement läuft.
Committed	Sie haben während der Testphase einen Plan gebucht. Der Plan startet nach dem Ende der Testphase.
Cancelled	Sie haben das Abonnement gekündigt.
Expired	Das Abonnement ist abgelaufen.

3.4.4.3 Next billing

Die Spalte **Next billing** zeigt die folgenden Werte an:

Wert	Bedeutung
`-`	Zeile der Testphase
`Activates 13.06.2026`	Der gebuchte Plan startet an diesem Datum.
`Ends 19.09.2026`	Das gekündigte Abonnement läuft bis zu diesem Datum weiter.
`Subscription cancelled`	Das gekündigte Abonnement ist beendet.
`19.09.2026`	Datum der nächsten Abrechnung

3.4.4.4 Options

Die Spalte **Options** enthält je nach Zustand des Eintrags die folgenden Schaltflächen:

Schaltfläche	Zeile	Wirkung
Explore Plans	Zeile der Testphase	Öffnet den Reiter **Plans**.

Schaltfläche	Zeile	Wirkung
Cancel commitment	Status **Committed**	Öffnet den Dialog **Cancel commitment**.
Manage	laufendes Abonnement	Öffnet das Kundenportal des Zahlungsdienstleisters. Dort verwalten Sie Zahlungsdaten und Rechnungen.
Cancel	laufendes Abonnement mit offener Abrechnung	Öffnet den Dialog **Cancel subscription**.

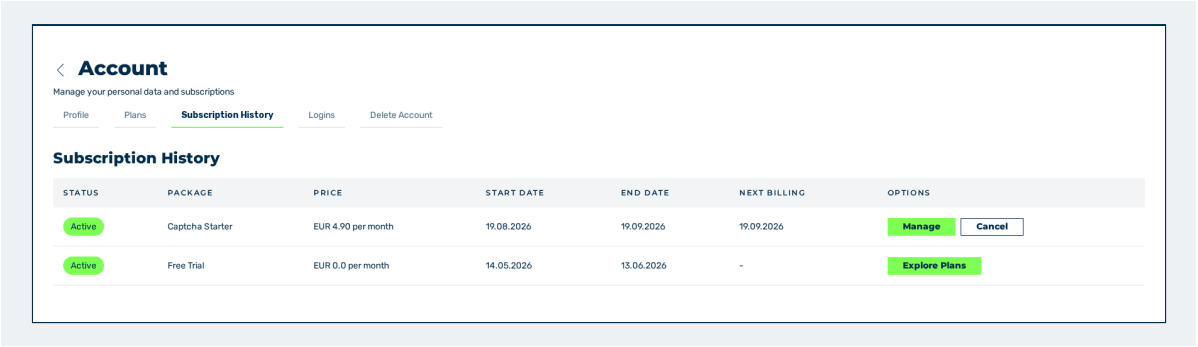


Abbildung 40: Reiter Subscription History mit einem gebuchten Plan

Siehe **Plan buchen**.

Siehe **Abonnement kündigen**.

3.4.5 Reiter Logins

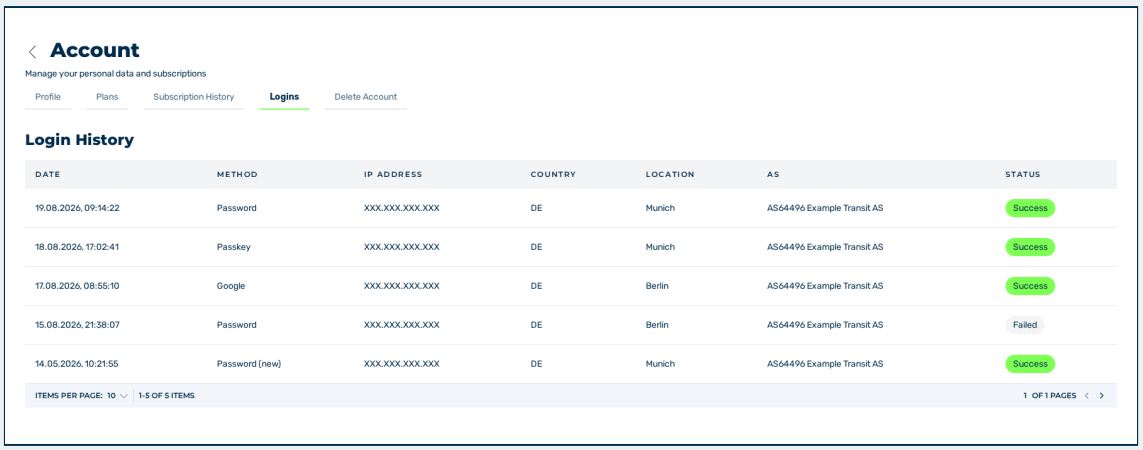


Abbildung 41: Reiter Logins

Der Reiter **Logins** protokolliert die Anmeldungen an Ihrem Konto. Aus dem Protokoll erkennen Sie, wann und von wo aus eine Anmeldung stattgefunden hat. Die Tabelle trägt die Überschrift **Login History**.

> **Video:** Das Protokoll der Anmeldungen — abspielbar in der Online-Hilfe

3.4.5.1 Spalten

Die Tabelle enthält die folgenden Spalten:

Spalte	Inhalt
Date	Zeitpunkt der Anmeldung im Format TT.MM.JJJJ, hh:mm:ss
Method	Verwendetes Anmeldeverfahren, zum Beispiel **Password**, **Passkey** oder **Google**. Die erste Anmeldung eines Kontos trägt den Zusatz **(new)**.
IP Address	IP-Adresse, von der die Anmeldung ausging
Country	Länderkennzeichen der IP-Adresse, zum Beispiel **DE**
Location	Ort der IP-Adresse. Ohne Ort steht hier das Land.

Spalte	Inhalt
AS	Autonomes System der IP-Adresse mit Nummer und Namen, zum Beispiel `AS64496 Example Transit AS`
Status	Ergebnis der Anmeldung als farbige Marke: **Success** oder **Failed**

Eine Spalte ohne Wert enthält ein `-`.

Unter der Tabelle blättern Sie durch das Protokoll und wählen die Anzahl der Einträge je Seite. Zu Beginn zeigt die Tabelle 10 Einträge je Seite.

Solange kein Eintrag vorliegt, zeigt die Tabelle den Hinweis **No login events** an.

Wenn das Protokoll eine Anmeldung enthält, die Sie nicht zuordnen können, dann ändern Sie umgehend Ihr Passwort und aktivieren Sie die Zwei-Faktor-Authentifizierung. Siehe **Passwort ändern**.

3.4.6 Reiter Delete Account

Abbildung 42: Reiter Delete Account

Der Reiter **Delete Account** löscht Ihr Konto endgültig. Die Schaltfläche **Delete Account Permanently** startet die Löschung.

> **Video:** Der Reiter Delete Account und die Löschung des Kontos — abspielbar in der Online-Hilfe

3.4.6.1 Folgen der Löschung

Die Löschung hat die folgenden Folgen:

- Ihr Konto und Ihre persönlichen Daten werden endgültig gelöscht.
- Alle Ihre Sitekeys und die zugehörigen CAPTCHA-Daten werden entfernt.
- Sie werden abgemeldet und können sich nicht erneut anmelden.

> Die Löschung lässt sich nicht rückgängig machen. Geschützte Websites verlieren ihren Schutz, sobald die Sitekeys entfernt sind.

3.4.6.2 Dialog Delete Account Permanently

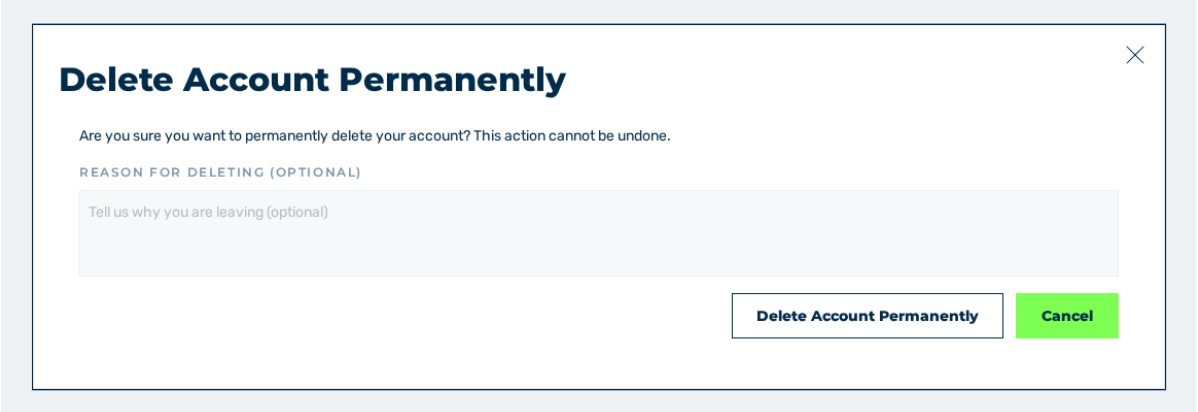


Abbildung 43: Dialog Delete Account Permanently

Die Schaltfläche **Delete Account Permanently** öffnet den gleichnamigen Dialog.

Das Feld **Reason for deleting (optional)** mit dem Platzhalter **Tell us why you are leaving (optional)** nimmt eine Begründung auf. Die Begründung ist freiwillig und fasst höchstens 2048 Zeichen.

Der Dialog enthält die folgenden Schaltflächen:

Schaltfläche	Wirkung
Delete Account Permanently	Löscht das Konto und meldet Sie ab.
Cancel	Schließt den Dialog, ohne zu löschen.

3.4.6.3 Bedingungen

Die Löschung ist erst möglich, wenn keine der folgenden Bedingungen zutrifft:

Bedingung	Titel des Dialogs	Maßnahme
Ein Abonnement ist aktiv.	**Cancel your subscription first**	Kündigen Sie das Abonnement. Die Schaltfläche **Go to Subscription History** öffnet den zugehörigen Reiter.
Eine Zahlung wird noch wiederholt.	**A payment is still being retried**	Warten Sie den Abschluss der Zahlung ab.

Die Schaltfläche **Close** schließt den Dialog.

Siehe **Reiter Subscription History**.

Siehe **Abonnement kündigen**.

3.5 Organisation

3.5.1 Organization

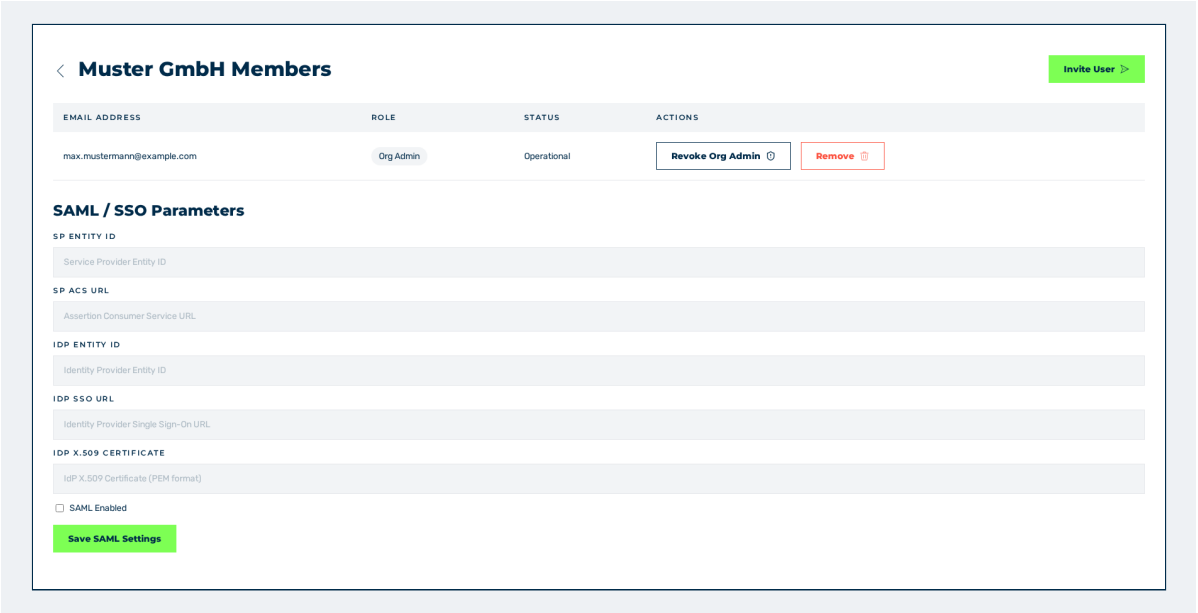


Abbildung 44: Ansicht Organization

Die Ansicht verwaltet die Mitglieder Ihrer Organisation und die Parameter für die Anmeldung über SAML und SSO. Die Ansicht steht Konten mit der Rolle **Org Admin** zur Verfügung. Die Überschrift nennt den Namen Ihrer Organisation, gefolgt von **Members**.

Sie öffnen die Ansicht im Reiter **Profile** über die Schaltfläche **Manage Members** im Bereich **Organization**.

3.5.1.1 Tabelle der Mitglieder

Die Tabelle enthält die folgenden Spalten:

Spalte	Inhalt
Email Address	E-Mail-Adresse des Mitglieds
Role	Rolle des Mitglieds als Marke
Status	Zustand des Kontos
Actions	Schaltflächen für den Eintrag

Solange kein Mitglied vorliegt, zeigt die Tabelle den Hinweis **No Members** an.

Die Schaltfläche **Invite User** neben der Überschrift öffnet den Dialog **Invite User**.

3.5.1.2 Rollen

Die folgenden Rollen stehen zur Verfügung:

Rolle	Rechte
Member	Nutzt das Produkt im Rahmen der eigenen Sitekeys.
Org Admin	Verwaltet zusätzlich die Mitglieder und die Einstellungen für SAML und SSO.

3.5.1.3 Status

Die Spalte **Status** nimmt einen der folgenden Werte an:

Wert	Bedeutung
Operational	Das Konto ist bestätigt und einsatzbereit.
Pending Verification	Die Einladung ist versendet, das Konto ist noch nicht bestätigt.

3.5.1.4 Aktionen

Die Spalte **Actions** enthält die folgenden Schaltflächen:

Schaltfläche	Wirkung
Make Org Admin	Öffnet den Dialog **Grant Org Admin**. Die Schaltfläche steht bei einem Mitglied mit der Rolle **Member**.
Revoke Org Admin	Öffnet den Dialog **Revoke Org Admin**. Die Schaltfläche steht bei einem Mitglied mit der Rolle **Org Admin**.
Remove	Öffnet den Dialog **Remove Member**.

Jeder der drei Dialoge nennt die E-Mail-Adresse des Mitglieds und enthält die Schaltflächen **Close** und **Confirm**.

Abbildung 45: Dialog Grant Org Admin

3.5.1.5 Bereich SAML / SSO Parameters

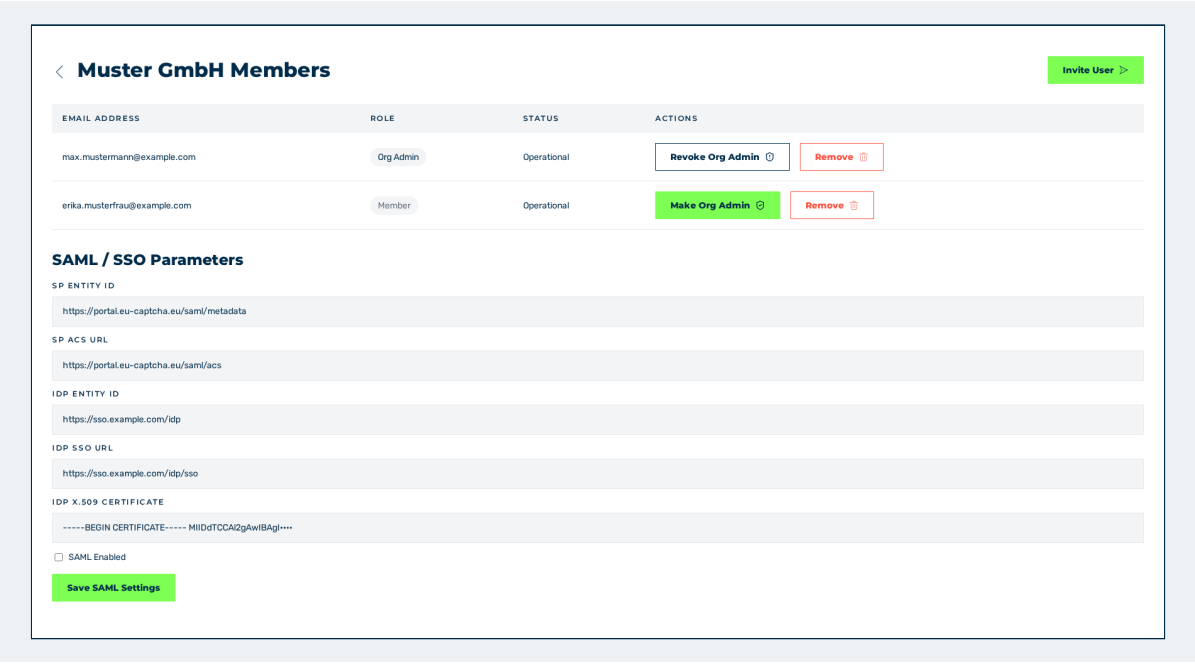


Abbildung 46: Bereich SAML / SSO Parameters

Der Bereich **SAML / SSO Parameters** verbindet Ihre Organisation mit einem Identitätsanbieter. Er enthält die folgenden Felder:

Feld	Inhalt
SP Entity ID	Kennung des Dienstanbieters, vorgegeben durch das Myra EU CAPTCHA
SP ACS URL	Adresse des Dienstanbieters für die Rückgabe der Anmeldung
IdP Entity ID	Kennung Ihres Identitätsanbieters

Feld	Inhalt
IdP SSO URL	Anmeldeadresse Ihres Identitätsanbieters
IdP X.509 Certificate	Zertifikat Ihres Identitätsanbieters im Format PEM

Das Kontrollkästchen **SAML Enabled** schaltet die Anmeldung über SAML ein. Die Schaltfläche **Save SAML Settings** übernimmt die Angaben. Nach dem Speichern erscheint die Meldung **SAML Settings** mit dem Text **SAML settings saved successfully.**

Siehe **Mitglied einladen**.

Siehe **SAML einrichten**.

3.6 Help & Support

3.6.1 Help

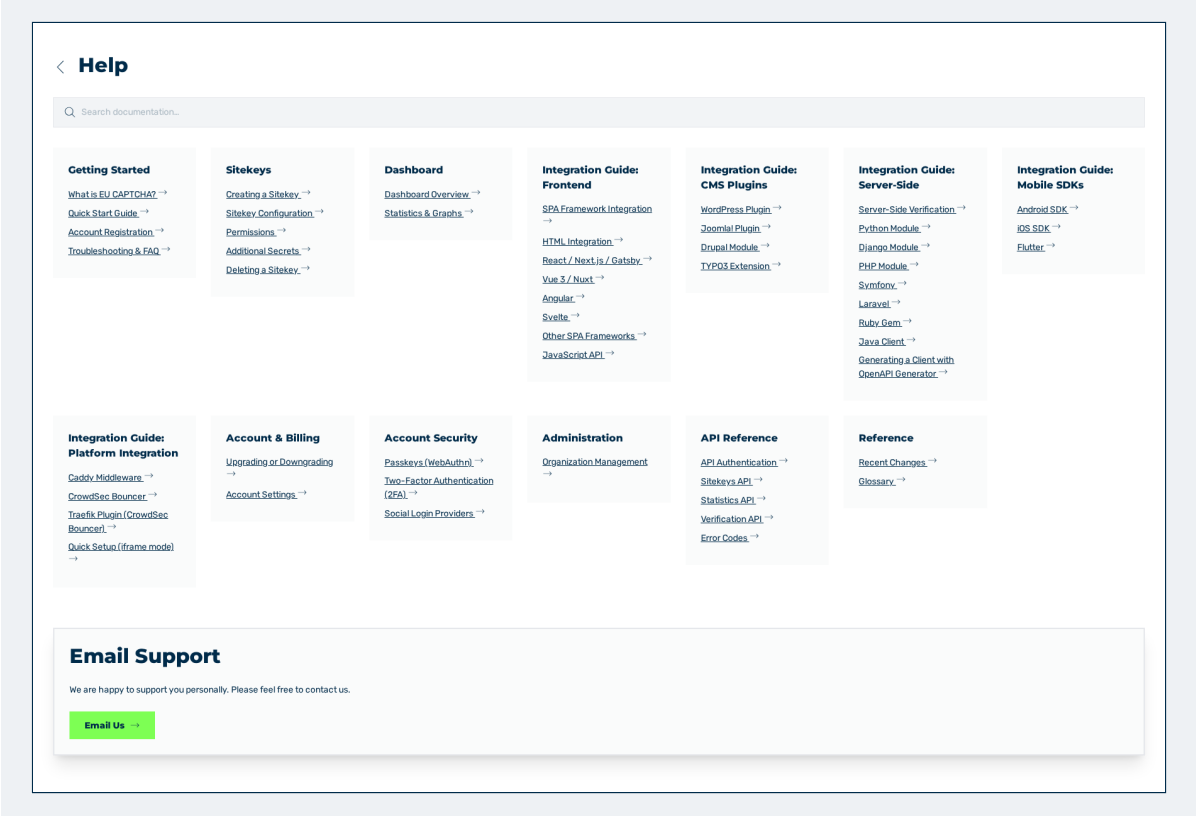


Abbildung 47: Ansicht Help

Die Ansicht **Help** führt zur Dokumentation und zum Support. Sie öffnen die Ansicht über den Eintrag **Help & Support** der Seitenleiste.

> **Video:** Die Ansicht Help und der Zugang zum Support — abspielbar in der Online-Hilfe

3.6.1.1 Suche

Am oberen Rand steht ein Suchfeld mit dem Platzhalter **Search documentation...**. Die Suche durchsucht die Dokumentation und zeigt die Treffer unter dem Feld an. Sie wählen einen Treffer mit den Pfeiltasten und der Eingabetaste aus.

Das Suchfeld steht auch in der Kopfzeile des Kundenportals zur Verfügung.

3.6.1.2 Themenkarten

Die Ansicht gliedert die Dokumentation in Karten. Die folgenden Karten stehen zur Verfügung:

Karte	Inhalt
Getting Started	Einstieg, Schnellstart, Registrierung, Fehlerbehebung und FAQ
Sitekeys	Anlegen und Verwalten von Sitekeys
Dashboard	Kennzahlen und Auswertungen
Integration Guide: Frontend	Einbindung in Frontend-Frameworks
Integration Guide: CMS Plugins	Erweiterungen für CMS-Plattformen
Integration Guide: Server-Side	Serverseitige Prüfung
Integration Guide: Mobile SDKs	Einbindung in mobile Apps
Integration Guide: Platform Integration	Einbindung in Proxys und Gateways
Account & Billing	Konto und Abrechnung
Account Security	Sicherheit des Kontos
Administration	Verwaltung der Organisation
API Reference	Beschreibung der Endpunkte
Reference	Nachschlagewerk

3.6.1.3 Support

Am unteren Rand steht die Karte **Email Support**. Die Karte enthält den Text **We are happy to support you personally. Please feel free to contact us.** und die Schaltfläche **Email Us**. Die Schaltfläche öffnet eine E-Mail an `eucaptcha.support@myrasecurity.com`.

Der Support steht in jedem Plan zur Verfügung.

Siehe **Reiter Plans**.

4. Konfiguration

4.1 Sitekey

4.1.1 Sitekey anlegen

Für jede Domain, die Sie schützen wollen, legen Sie einen eigenen Sitekey an. Der Sitekey verbindet die Domain mit dem Myra EU CAPTCHA und liefert die Schlüssel für das Widget und für die serverseitige Prüfung.

> **Video:** Anlegen eines weiteren Sitekeys — abspielbar in der Online-Hilfe

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet.

☑ Sie kennen die Domain, die Sie schützen wollen.

Um einen weiteren Sitekey anzulegen, gehen Sie wie folgt vor:

Abbildung 48: Bereich Protect your website

- Öffnen Sie die Ansicht **Dashboard** und blättern Sie zum Bereich **Protect your website**. Die Beschreibung lautet **Takes 30 seconds. Enter the domain where you want to add EU CAPTCHA.**
- Geben Sie in das Feld **Domain** die zu schützende Domain ein, zum Beispiel `example.com`.
- Falls erforderlich, geben Sie in das Feld **Label** eine Bezeichnung ein, zum Beispiel `Production`, `Staging` oder `My blog`.
- Klicken Sie auf die Schaltfläche **Create Sitekey**.

↳ Das System legt den Sitekey an und zeigt die Meldung **Sitekey created!** an. Der Text lautet **Now add it to your website – it only takes a minute.** Der neue Eintrag erscheint in der Liste **Sitekey List**.

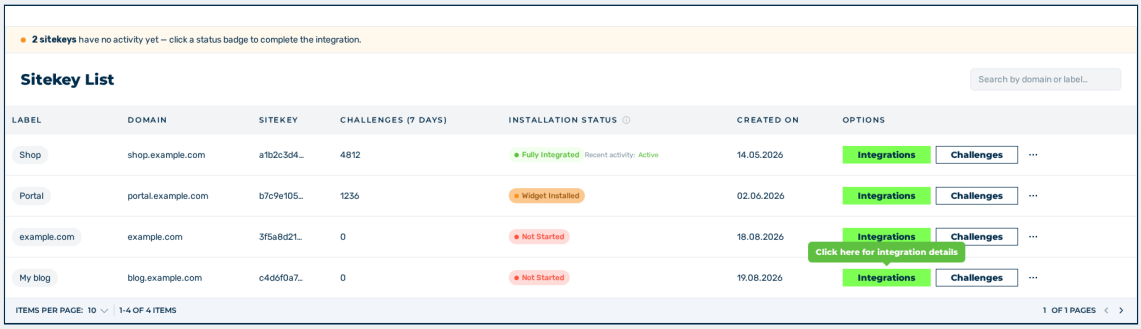


Abbildung 49: Bereich Sitekey List mit dem neuen Eintrag

► Klicken Sie auf die Schaltfläche **Set up integration**.

➔ Die Ansicht **Details** des neuen Sitekeys wird angezeigt.

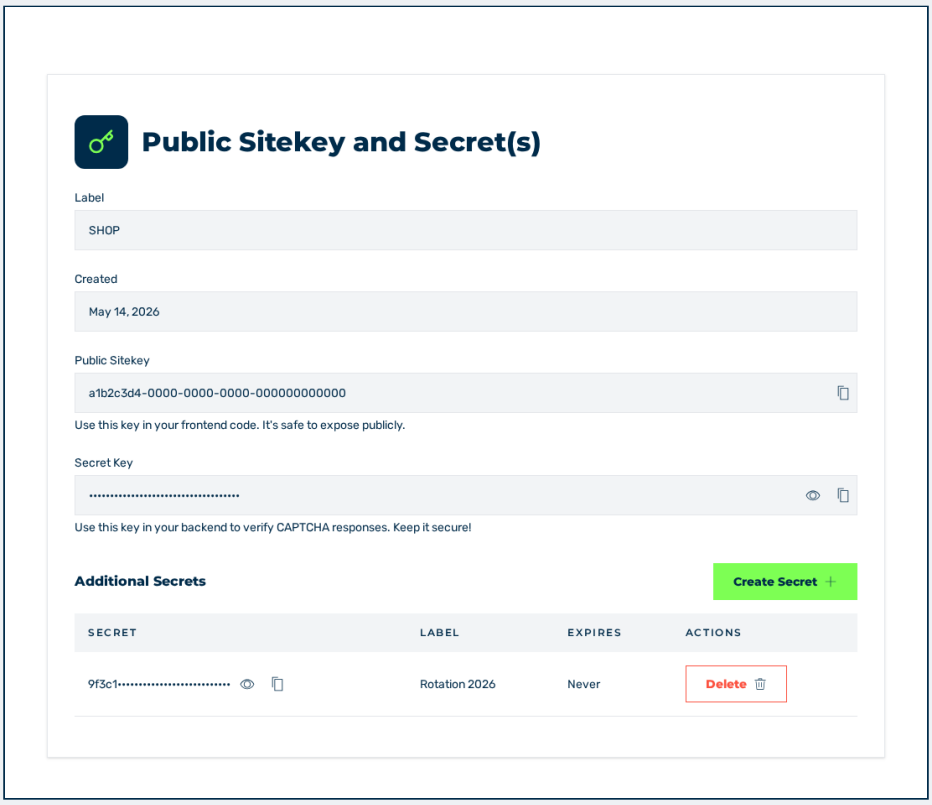


Abbildung 50: Ansicht Details

4.1.1.1 Prüfung der Eingabe

Das Feld **Domain** weist fehlerhafte Eingaben zurück:

Meldung	Ursache
Please enter a domain.	Das Feld ist leer.
Please enter a valid domain name (e.g. example.com).	Die Eingabe ist kein gültiger Domainname.

Eine führende Angabe `http://` oder `https://` sowie ein angehängter Pfad werden aus der Eingabe entfernt. Groß- und Kleinschreibung spielen keine Rolle: Das System speichert die Domain in Kleinbuchstaben.

Das System verhindert derzeit nicht, dass mehrere Sitekeys für dieselbe Domain entstehen. Prüfen Sie die Liste und löschen Sie doppelte Einträge. Siehe **Doppelter Sitekey für dieselbe Domain**.

Siehe **Sitekey konfigurieren**.

4.1.2 Sitekey konfigurieren

Die Konfiguration steuert, wie streng das Myra EU CAPTCHA die Besucher einer Domain prüft. Sie schalten den Schutz ein und aus. Zusätzlich legen Sie Anfangsschwierigkeit, Wartezeit und die Zahl paralleler Challenges fest.

> **Video:** Ändern der Konfiguration eines Sitekeys — abspielbar in der Online-Hilfe

Die folgenden Voraussetzungen müssen erfüllt sein:

- ☑ Sie haben die Zugriffsstufe **Write** auf den Sitekey.
- ☑ Für Ihr Konto ist ein kostenpflichtiger Plan aktiv. Im Tarif **free** sind die drei Zahlenwerte gesperrt. Das Feld **Label** und der Schalter **EU CAPTCHA Protection** bleiben offen.

Die Ansicht kennt keine Schaltfläche zum Speichern. Das System übernimmt jede Änderung sofort und meldet **Success** mit dem Text **Sitekey config updated successfully**.

Um einen Sitekey zu konfigurieren, gehen Sie wie folgt vor:

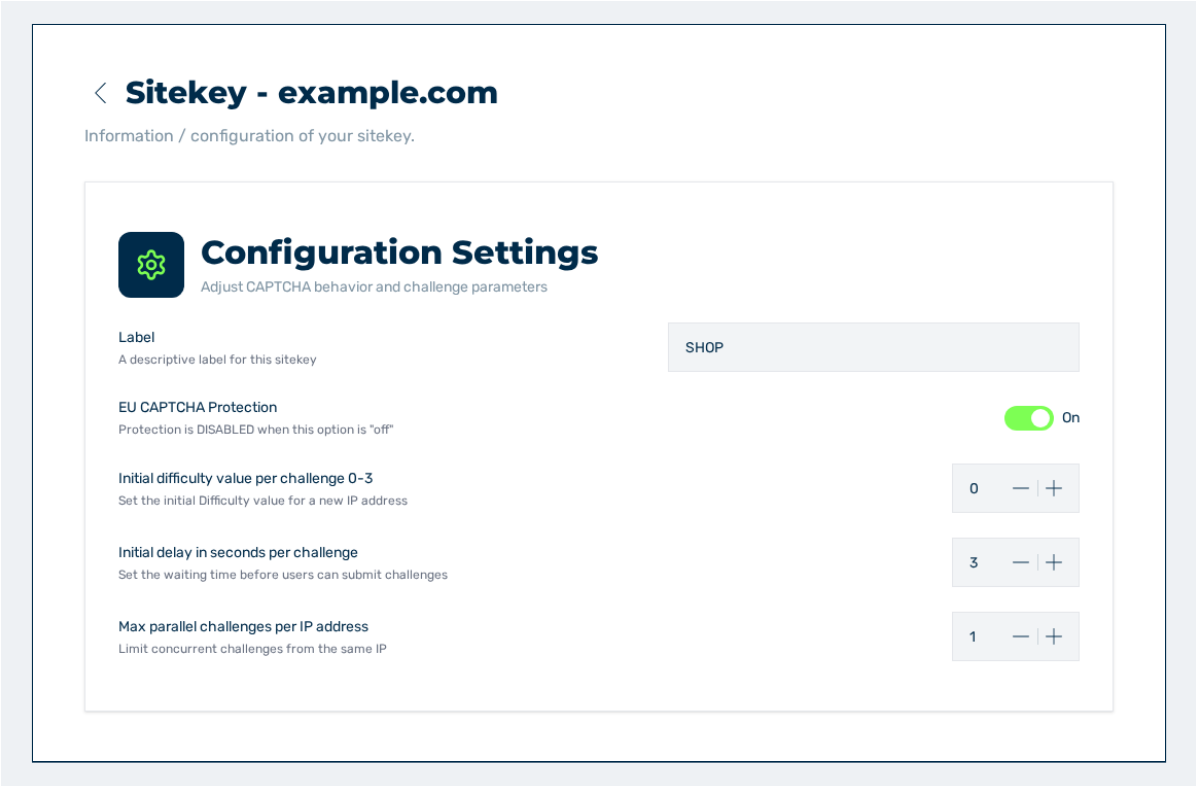


Abbildung 51: Ansicht Configuration

- ▶ Öffnen Sie den Sitekey und wechseln Sie in die Ansicht **Configuration**.
- ▶ Falls erforderlich, geben Sie in das Feld **Label** eine Bezeichnung ein und drücken Sie die Eingabetaste.
- ▶ Stellen Sie den Schalter **EU CAPTCHA Protection** auf **On**.
- ▶ Geben Sie in das Feld **Initial difficulty value per challenge 0-3** einen Wert zwischen 0 und 3 ein.
- ▶ Geben Sie in das Feld **Initial delay in seconds per challenge** einen Wert zwischen 3 und 30 ein.
- ▶ Geben Sie in das Feld **Max parallel challenges per IP address** einen Wert zwischen 1 und 10 ein.
- ➔ Das System übernimmt jede Änderung sofort und wendet die Werte auf die folgenden Challenges an.

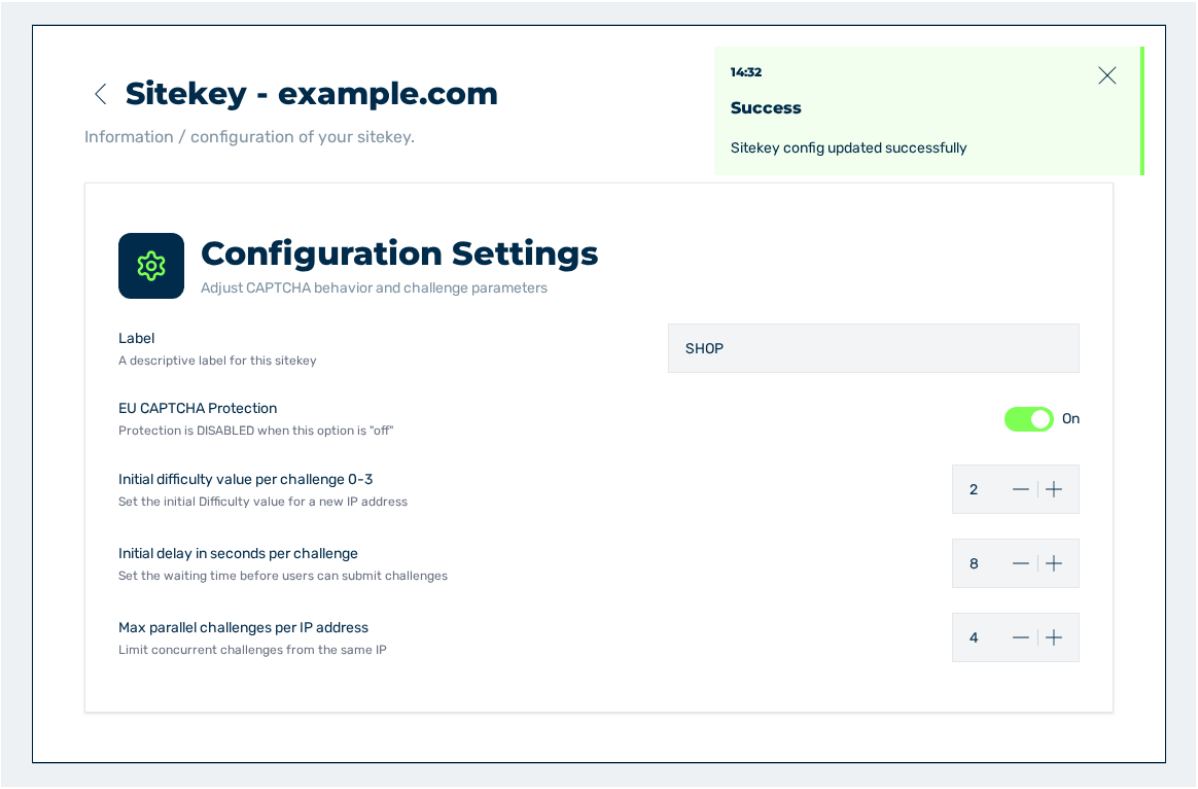


Abbildung 52: Ansicht Configuration mit den geänderten Werten

4.1.2.1 Wirkung der Einstellungen

Die Einstellungen wirken wie folgt:

Einstellung	Wirkung	Vorgabe
EU CAPTCHA Protection	Auf **Off** stellt das Widget keine Challenge mehr, die Anfragen laufen ungeprüft durch.	**On**
Initial difficulty value per challenge 0-3	Legt die Schwierigkeit der ersten Challenge einer noch unbekannten IP-Adresse fest. Ein höherer Wert verlängert die Prüfung für alle Besucher.	0
Initial delay in seconds per challenge	Legt die Wartezeit fest, bevor das Absenden möglich ist.	3
	Begrenzt die gleichzeitigen Challenges einer IP-Adresse.	1

Einstellung	Wirkung	Vorgabe
Max parallel challenges per IP address		

Die höheren Schwierigkeitsgrade bis Stufe 7 vergibt die Risikoerkennung selbst. Siehe **Funktionsweise**.

> Auf **Off** besteht kein Schutz. Das Widget bleibt eingebunden, die Anfragen werden jedoch nicht mehr geprüft.

4.1.2.2 Gesperrte Einstellungen

Wenn die Einstellungen gesperrt sind, dann nennt ein Kurzhinweis den Grund:

Kurzhinweis	Ursache	Maßnahme
Read-only access	Ihre Zugriffsstufe ist **Read**.	Lassen Sie sich die Stufe **Write** erteilen. Siehe **Zugriff auf einen Sitekey ändern**.
Upgrade to unlock	Für Ihr Konto ist kein kostenpflichtiger Plan aktiv.	Buchen Sie einen Plan. Siehe **Plan buchen**.

4.1.3 Zusätzliche Secrets verwalten

Neben dem ursprünglichen Secret legen Sie zu einem Sitekey weitere Secrets an. Damit wechseln Sie den Schlüssel im laufenden Betrieb, ohne die Prüfung zu unterbrechen: Sie legen ein zweites Secret an, stellen Ihre Server nacheinander darauf um und löschen anschließend das alte.

4.1.3.1 Secret anlegen

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben die Zugriffsstufe **Write** auf den Sitekey.

Um ein zusätzliches Secret anzulegen, gehen Sie wie folgt vor:

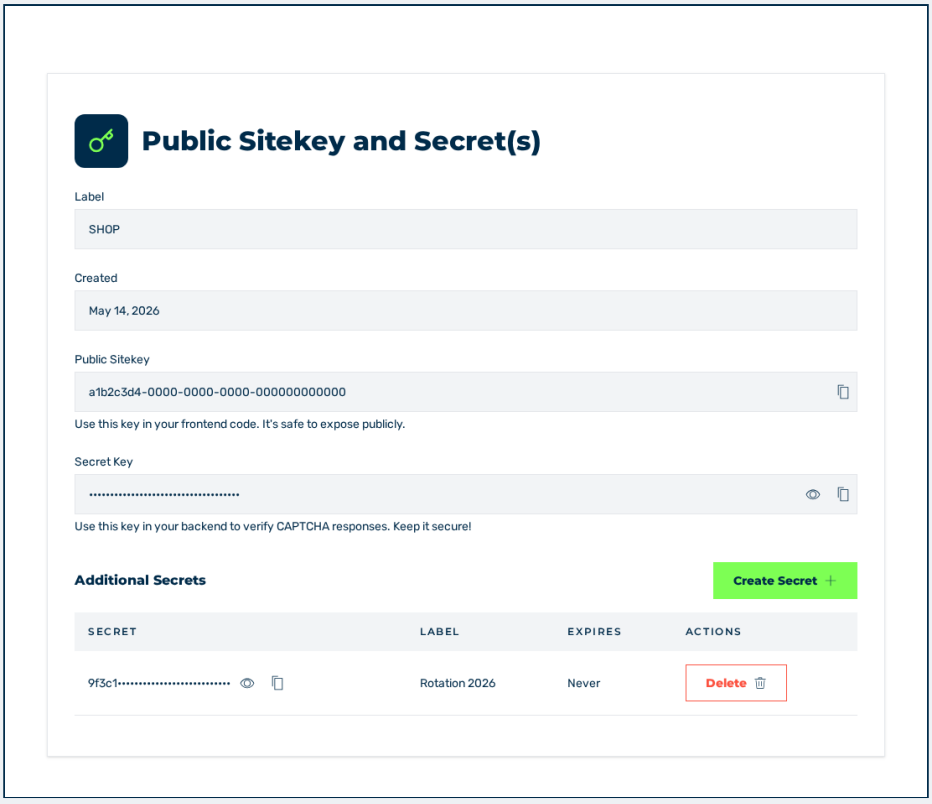


Abbildung 53: Ansicht Details

- ► Öffnen Sie den Sitekey und wechseln Sie in die Ansicht **Details**.
- ► Klicken Sie im Bereich **Additional Secrets** auf die Schaltfläche **Create Secret**.
  - ↳ Der Dialog **Create Secret** öffnet sich.

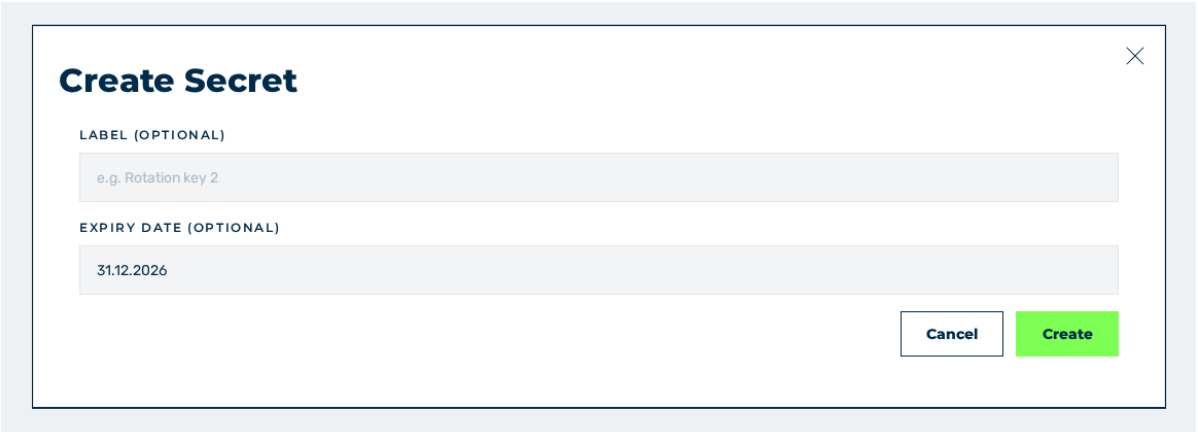


Abbildung 54: Dialog Create Secret

- ▶ Falls erforderlich, geben Sie in das Feld **Label (optional)** eine Bezeichnung ein, zum Beispiel `Rotation key 2`.
- ▶ Falls erforderlich, wählen Sie im Feld **Expiry date (optional)** ein Ablaufdatum aus.
- ▶ Klicken Sie auf die Schaltfläche **Create**.
- ➔ Das System legt das Secret an und zeigt es im Bereich **Additional Secrets** an.

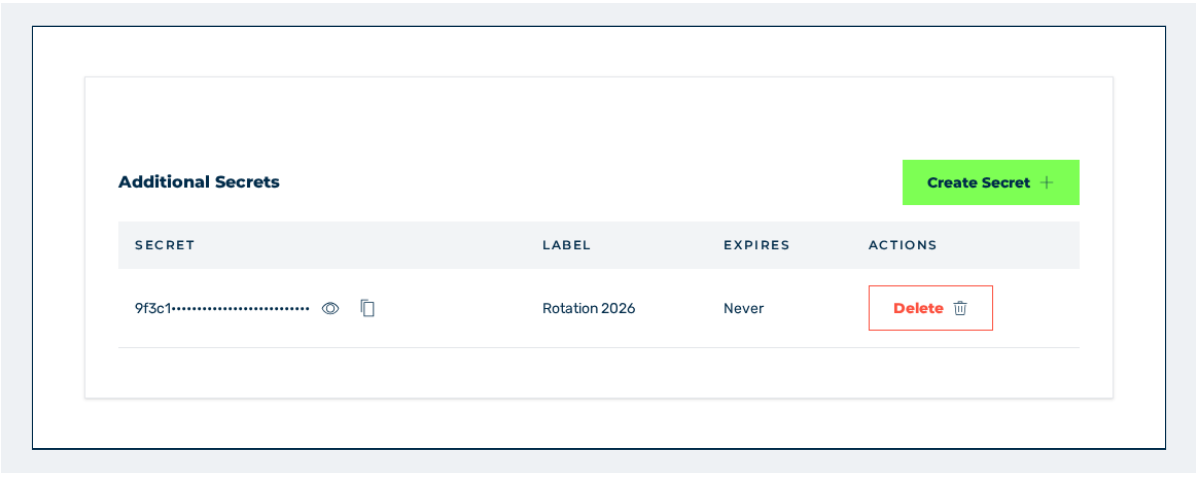


Abbildung 55: Bereich Additional Secrets mit dem neuen Secret

Ohne Ablaufdatum läuft das Secret nicht ab. Die Spalte **Expires** zeigt in diesem Fall den Wert **Never** an.

Um den Vorgang abzubrechen, klicken Sie im Dialog auf die Schaltfläche **Cancel**.

Behandeln Sie jedes zusätzliche Secret wie das ursprüngliche Secret. Es gehört ausschließlich auf Ihren Server und niemals in HTML oder JavaScript.

4.1.3.2 Secret löschen

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben die Zugriffsstufe **Write** auf den Sitekey.

☑ Keiner Ihrer Server verwendet das Secret noch.

Um ein zusätzliches Secret zu löschen, gehen Sie wie folgt vor:

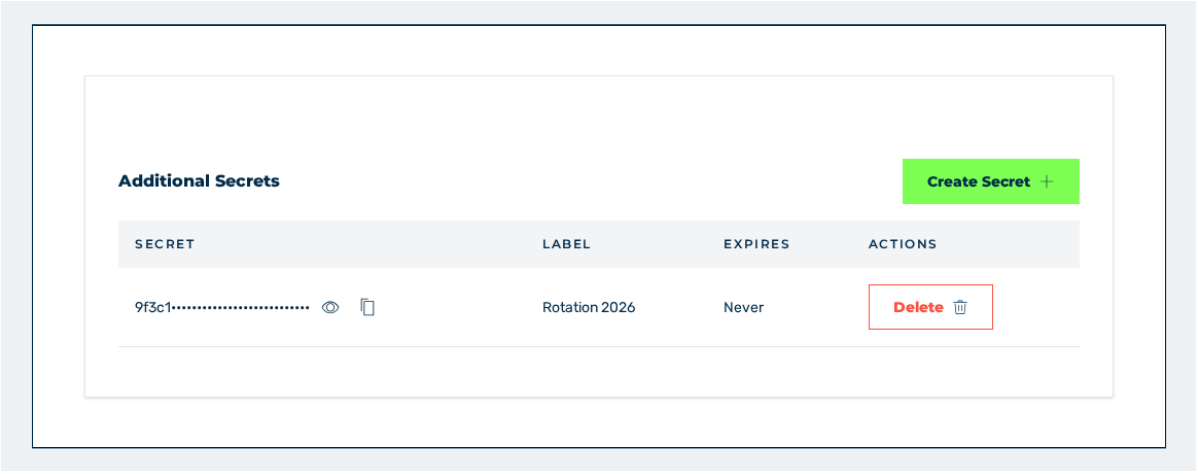


Abbildung 56: Bereich Additional Secrets

► Klicken Sie im Bereich **Additional Secrets** in der Zeile des Secrets auf die Schaltfläche **Delete**.

↳ Der Dialog **Delete Secret** öffnet sich.

Abbildung 57: Dialog Delete Secret

► Klicken Sie auf die Schaltfläche **Delete**.

➔ Das System löscht das Secret und zeigt die Meldung **Secret deleted** mit dem Text **The secret has been removed** an.

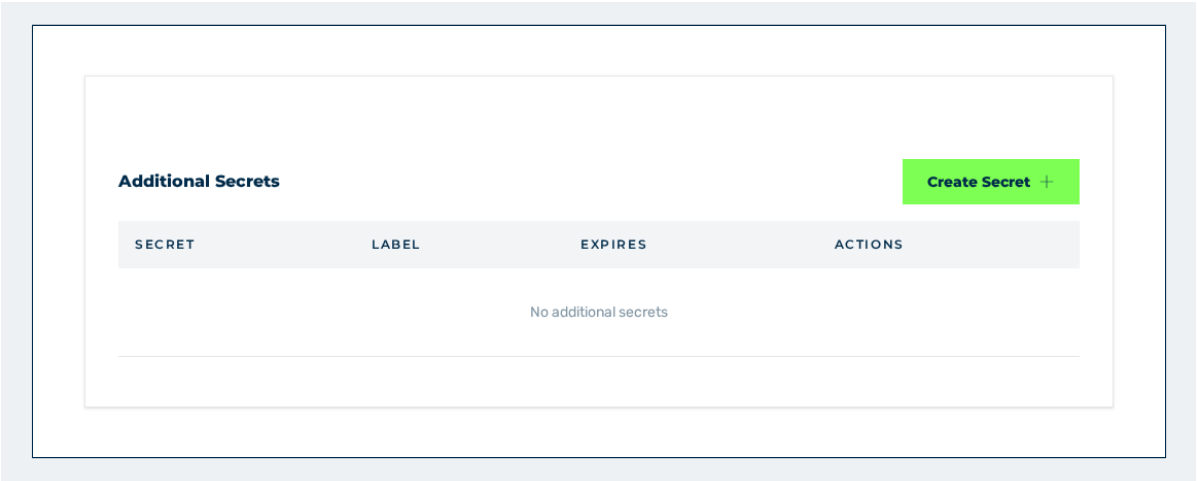


Abbildung 58: Bereich Additional Secrets ohne zusätzliche Secrets

Um den Vorgang abzubrechen, klicken Sie im Dialog auf die Schaltfläche **Cancel**.

Ein gelöschtes Secret lässt sich nicht wiederherstellen. Anfragen, die noch mit diesem Secret geprüft werden, schlagen ab sofort fehl.

Siehe **Details**.

4.1.4 Sitekey löschen

Ein gelöschter Sitekey schützt seine Domain nicht mehr. Löschen Sie einen Sitekey erst, wenn Sie das Widget aus der zugehörigen Website entfernt haben.

> **Video:** Löschen eines Sitekeys — abspielbar in der Online-Hilfe

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben die Zugriffsstufe **Write** auf den Sitekey.

☑ Das Widget ist aus der geschützten Website entfernt, oder die Website wird nicht mehr betrieben.

Um einen Sitekey zu löschen, gehen Sie wie folgt vor:

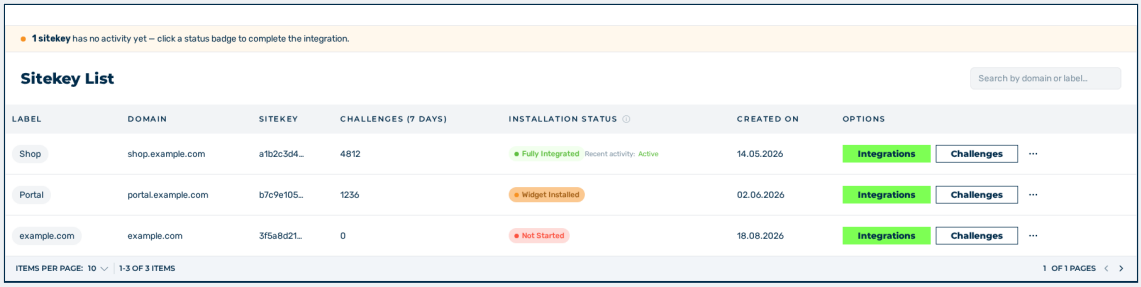


Abbildung 59: Bereich Sitekey List

► Öffnen Sie die Ansicht **Dashboard** und blättern Sie zum Bereich **Sitekey List**.

► Klicken Sie in der Zeile des Sitekeys in der Spalte **Options** auf die Aktion zum Löschen.

↳ Der Dialog **Delete Sitekey?** wird angezeigt.

Abbildung 60: Dialog Delete Sitekey?

► Klicken Sie auf die Schaltfläche **Delete**.

➔ Das System löscht den Sitekey. Der Eintrag verschwindet aus der Liste.

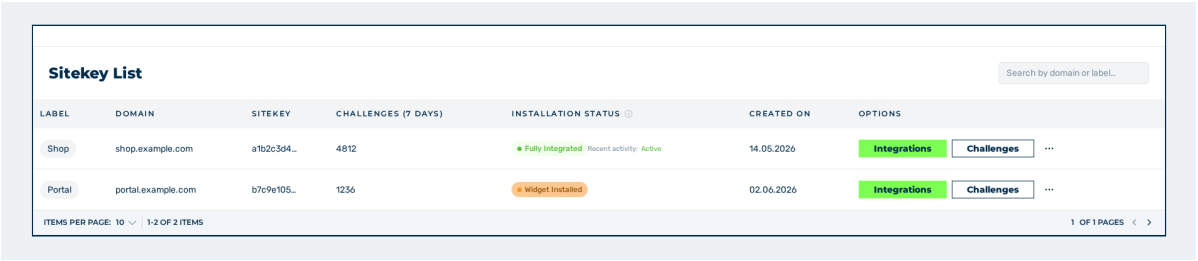


Abbildung 61: Bereich Sitekey List ohne den gelöschten Eintrag

Nach dem Löschen weist das Widget den Sitekey zurück. Formulare, die den Schlüssel noch verwenden, erhalten kein gültiges Token mehr.

Um den Vorgang abzubrechen, klicken Sie im Dialog auf die Schaltfläche **Cancel**.

4.2 Berechtigungen

4.2.1 Zugriff auf einen Sitekey gewähren

Sie geben einzelnen Konten Zugriff auf einen Sitekey, ohne Ihre eigenen Zugangsdaten weiterzugeben. Der Zugriff gilt jeweils für einen Sitekey.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben die Zugriffsstufe **Write** auf den Sitekey.

☑ Das Konto, das Zugriff erhalten soll, ist im Kundenportal registriert.

☑ Sie kennen die E-Mail-Adresse dieses Kontos.

Um Zugriff auf einen Sitekey zu gewähren, gehen Sie wie folgt vor:

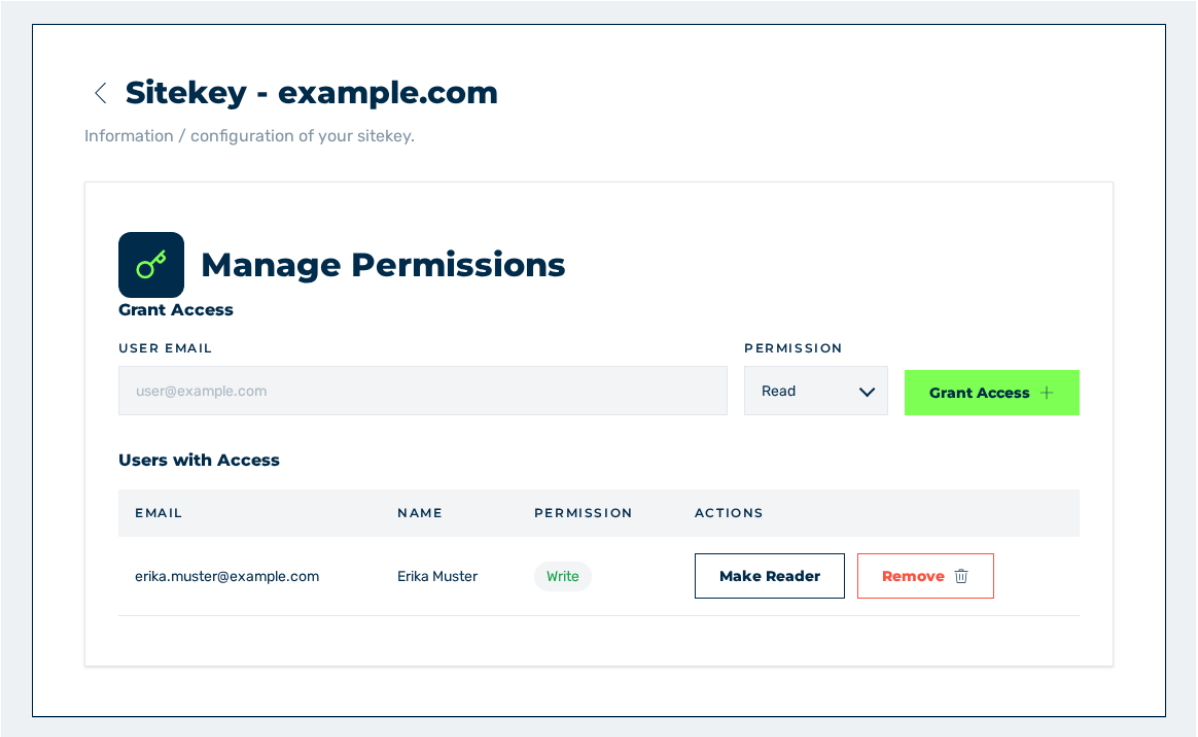


Abbildung 62: Ansicht Permissions

► Öffnen Sie den Sitekey und wechseln Sie in die Ansicht **Permissions**.

► Geben Sie im Bereich **Grant Access** in das Feld **User Email** die E-Mail-Adresse des Kontos ein.

- ► Wählen Sie in der Drop-down-Liste **Permission** die Zugriffsstufe **Read** oder **Write** aus.
- ► Klicken Sie auf die Schaltfläche **Grant Access**.
- ➔ Das System vergibt den Zugriff und zeigt die Meldung **Permission granted** an. Der Eintrag erscheint im Bereich **Users with Access**.

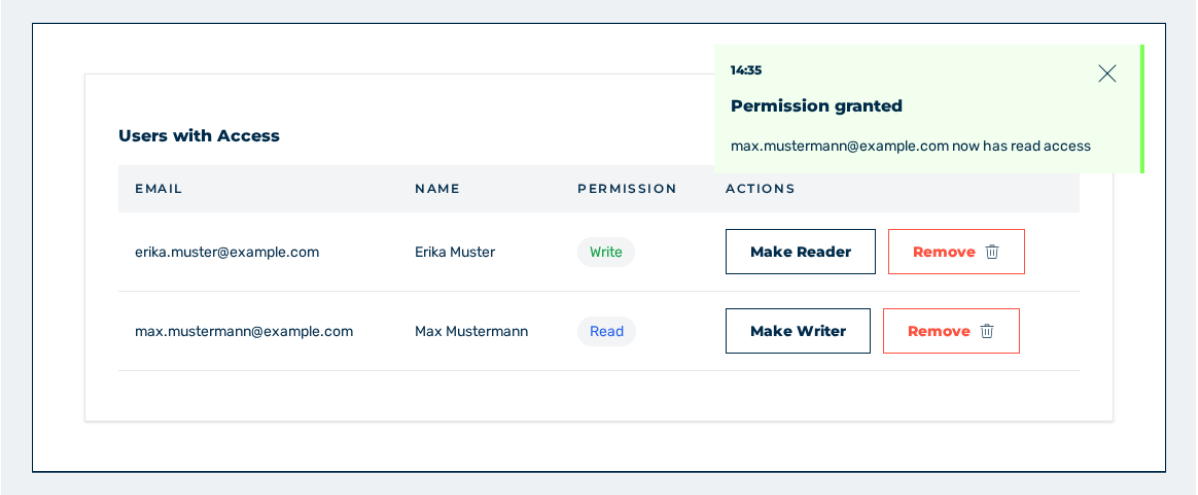


Abbildung 63: Bereich Users with Access mit dem neuen Eintrag

Die folgenden Zugriffsstufen stehen zur Verfügung:

Stufe	Rechte
Read	Der Sitekey und seine Einstellungen sind sichtbar. Änderungen sind gesperrt.
Write	Der Sitekey und seine Einstellungen sind sichtbar und änderbar.

Siehe **Zugriff auf einen Sitekey ändern**.

4.2.2 Zugriff auf einen Sitekey ändern

Sie ändern die Zugriffsstufe eines berechtigten Kontos zwischen **Read** und **Write**.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben die Zugriffsstufe **Write** auf den Sitekey.

☑ Das Konto ist bereits berechtigt. Siehe **Zugriff auf einen Sitekey gewähren**.

Um die Zugriffsstufe zu ändern, gehen Sie wie folgt vor:

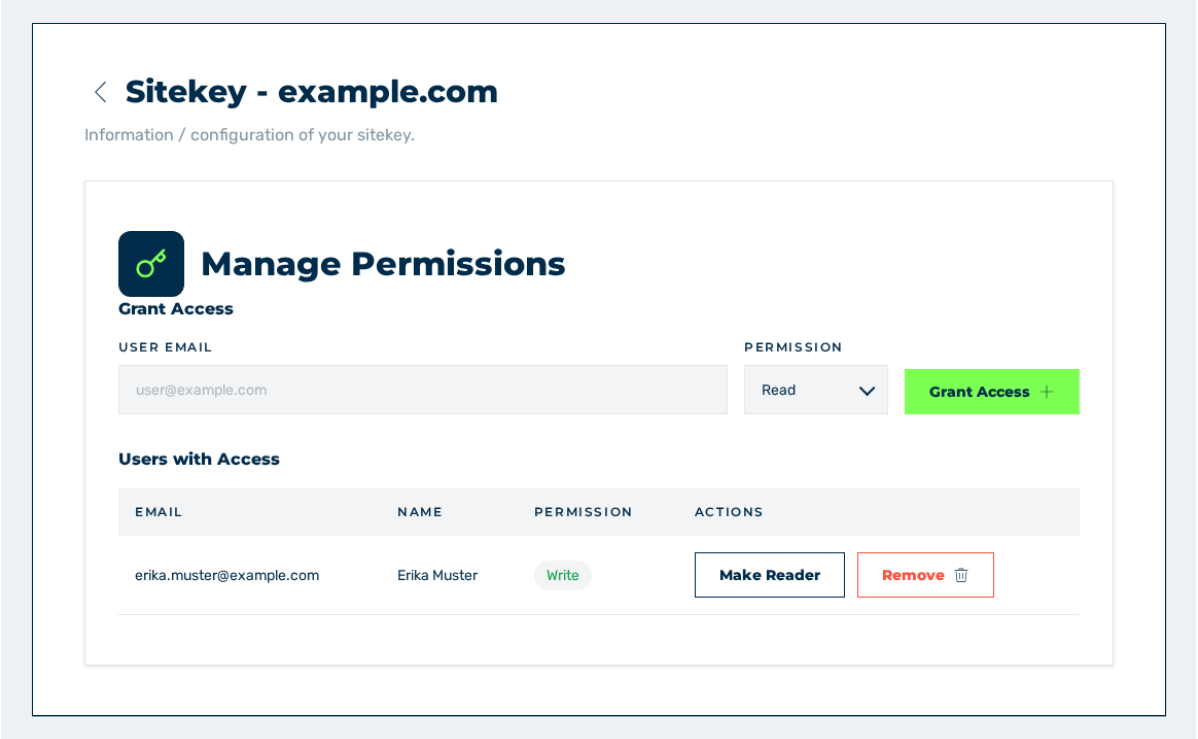


Abbildung 64: Bereich Users with Access

► Öffnen Sie den Sitekey und wechseln Sie in die Ansicht **Permissions**.

► Klicken Sie im Bereich **Users with Access** in der Zeile des Kontos auf die Aktion zum Wechsel der Stufe.

↳ Der Dialog **Change to Write** oder **Change to Read** wird angezeigt. Die Frage nennt das Konto und beide Stufen.

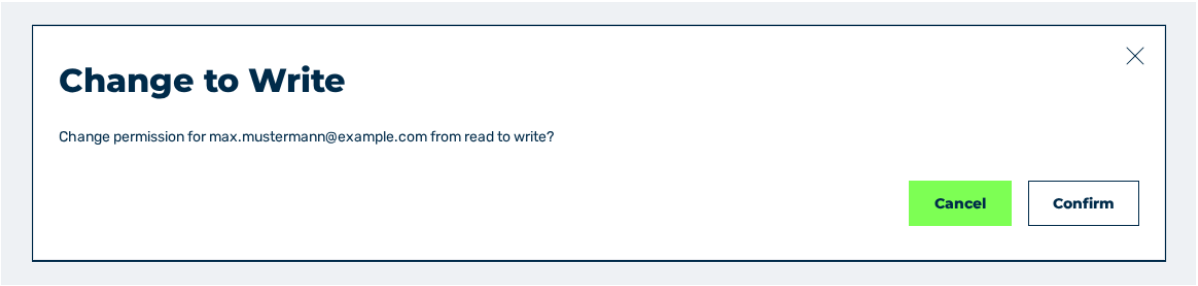


Abbildung 65: Dialog Change to Write

- ► Klicken Sie auf die Schaltfläche **Confirm**.
- ➔ Das System ändert die Stufe und zeigt die Meldung **Permission updated** an. Die Spalte **Permission** zeigt die neue Stufe an.

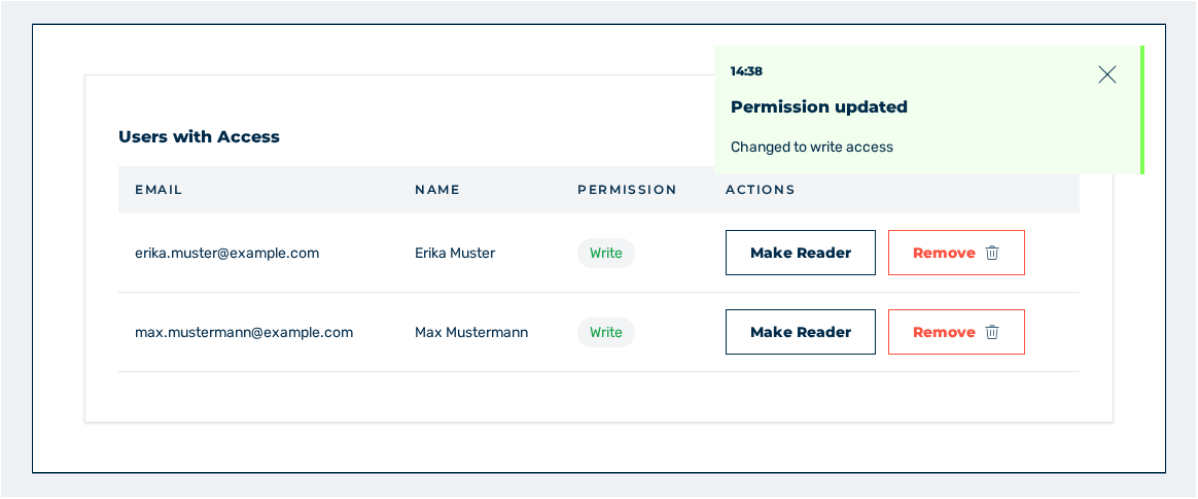


Abbildung 66: Bereich Users with Access mit der geänderten Stufe

Um den Vorgang abzubrechen, klicken Sie im Dialog auf die Schaltfläche **Cancel**.

4.2.3 Zugriff auf einen Sitekey entziehen

Sie entziehen einem Konto den Zugriff auf einen Sitekey. Der Sitekey selbst bleibt bestehen.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben die Zugriffsstufe **Write** auf den Sitekey.

☑ Das Konto ist berechtigt.

Um den Zugriff zu entziehen, gehen Sie wie folgt vor:

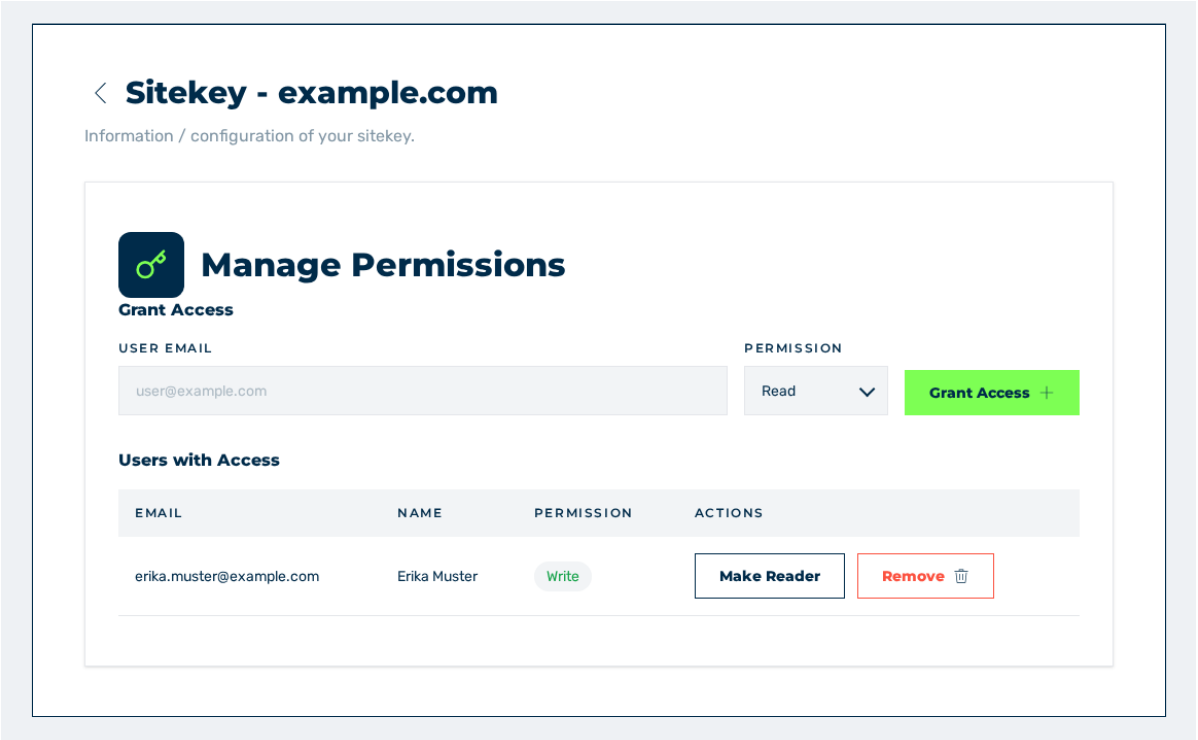


Abbildung 67: Bereich Users with Access

► Öffnen Sie den Sitekey und wechseln Sie in die Ansicht **Permissions**.

► Klicken Sie im Bereich **Users with Access** in der Zeile des Kontos auf die Aktion **Remove**.

↳ Der Dialog **Remove Permission** wird angezeigt. Die Frage nennt die E-Mail-Adresse des Kontos.

Abbildung 68: Dialog Remove Permission

- ► Klicken Sie auf die Schaltfläche **Confirm**.
- ➔ Das System entzieht den Zugriff und zeigt die Meldung **Permission revoked** an. Der Eintrag verschwindet aus der Liste.

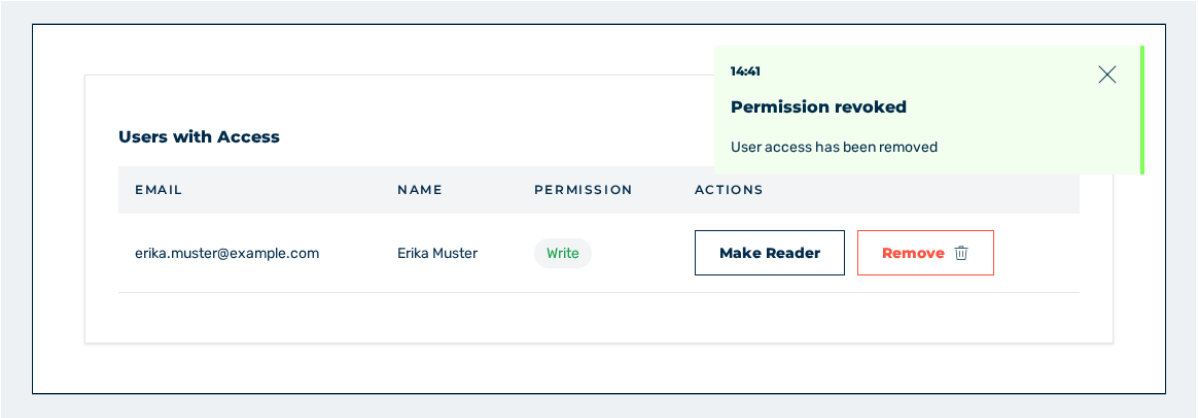


Abbildung 69: Bereich Users with Access ohne den Eintrag

Um den Vorgang abzubrechen, klicken Sie im Dialog auf die Schaltfläche **Cancel**.

4.3 Konto und Sicherheit

4.3.1 Persönliche Daten ändern

Vorname, Nachname und Firmenname stehen im Bereich **Profile** des gleichnamigen Reiters. Die E-Mail-Adresse ändern Sie dort nicht direkt: Ihr Feld ist gesperrt.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet.

Um die persönlichen Daten zu ändern, gehen Sie wie folgt vor:

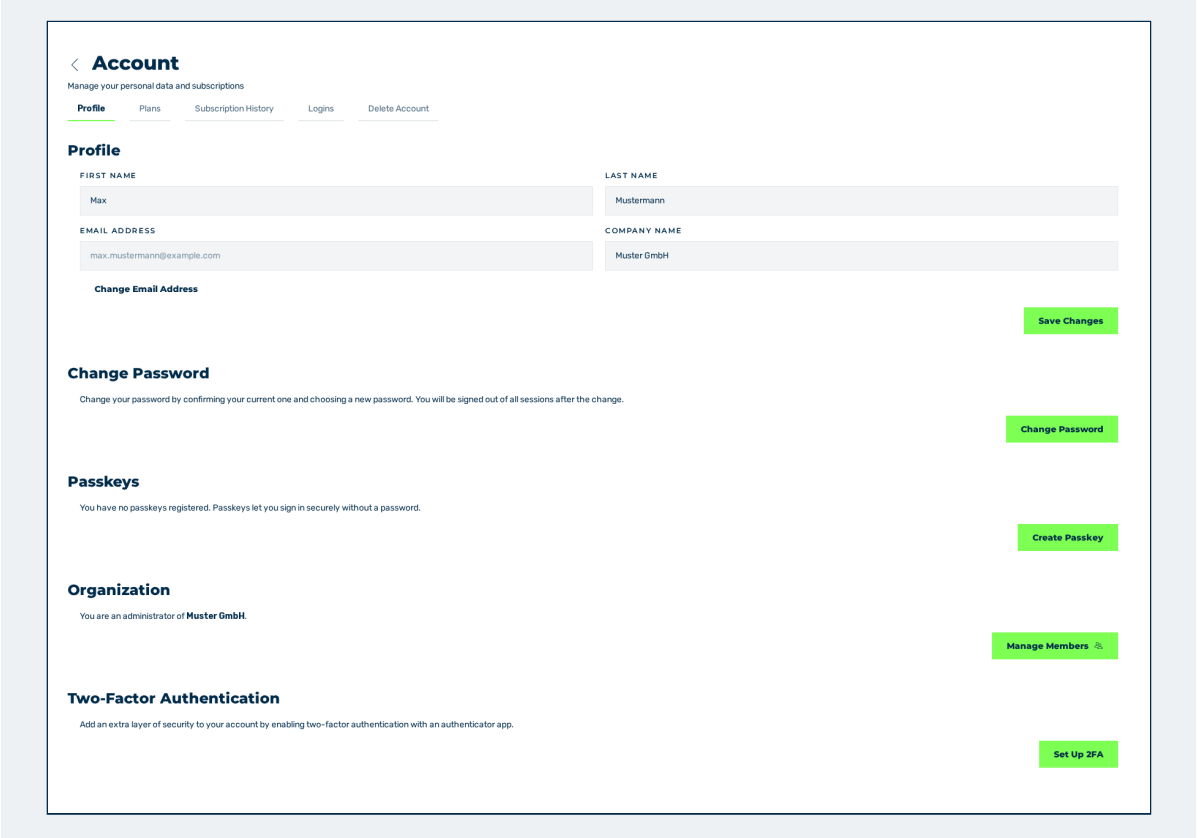


Abbildung 70: Reiter Profile

- ▶ Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile**.
- ▶ Geben Sie in das Feld **First name** Ihren Vornamen ein.
- ▶ Geben Sie in das Feld **Last name** Ihren Nachnamen ein.

- ► Falls erforderlich, geben Sie in das Feld **Company name** den Namen Ihres Unternehmens ein.
- ► Klicken Sie auf die Schaltfläche **Save Changes**.
- ➔ Das System speichert die Daten.

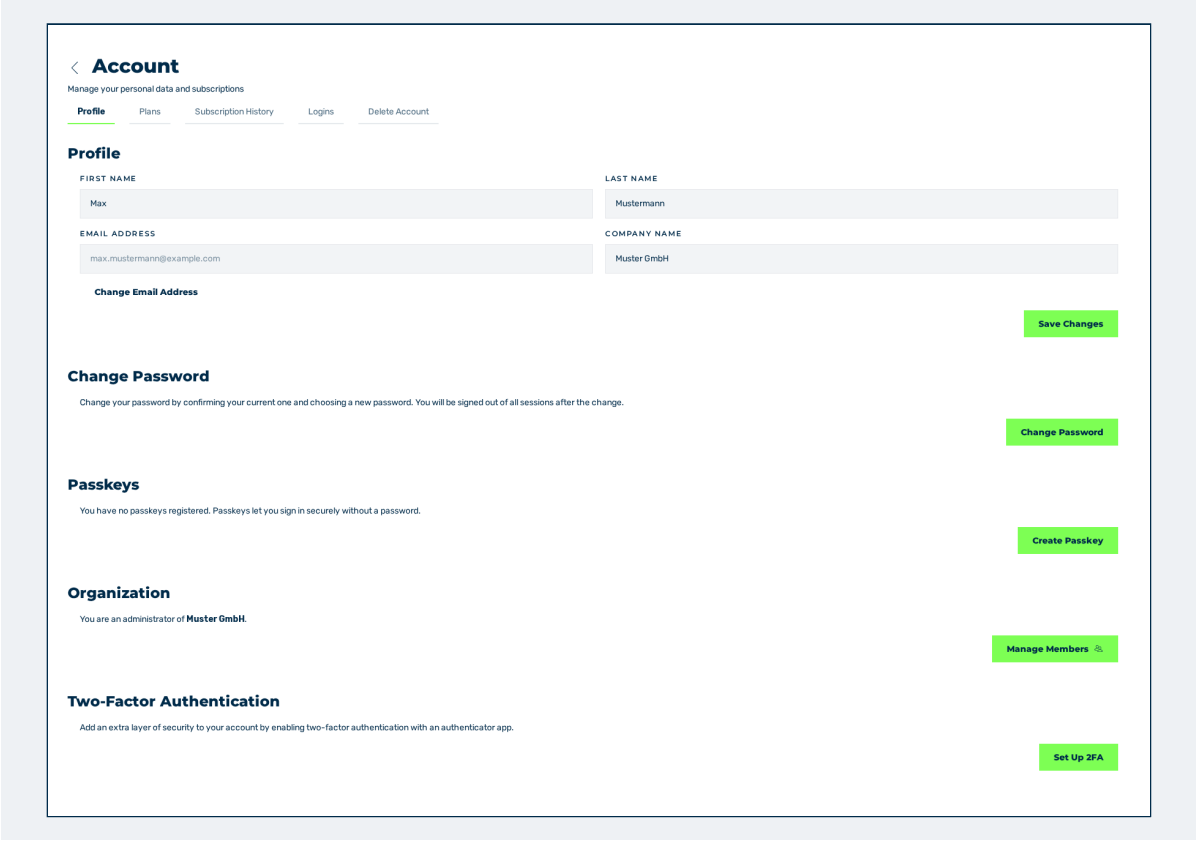


Abbildung 71: Reiter Profile mit den gespeicherten Daten

Das Feld **Email address** nennt die E-Mail-Adresse Ihres Kontos. Es ist gesperrt und lässt sich nur über die Schaltfläche **Change Email Address** ändern. Siehe **E-Mail-Adresse ändern**.

Siehe **Reiter Profile**.

4.3.2 E-Mail-Adresse ändern

Sie ändern die E-Mail-Adresse Ihres Kontos im Reiter **Profile**. Die Änderung wird erst wirksam, wenn Sie sie über die bisherige Adresse bestätigen.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet.

☑ Sie haben Zugriff auf das Postfach der bisherigen E-Mail-Adresse.

Um die E-Mail-Adresse zu ändern, gehen Sie wie folgt vor:

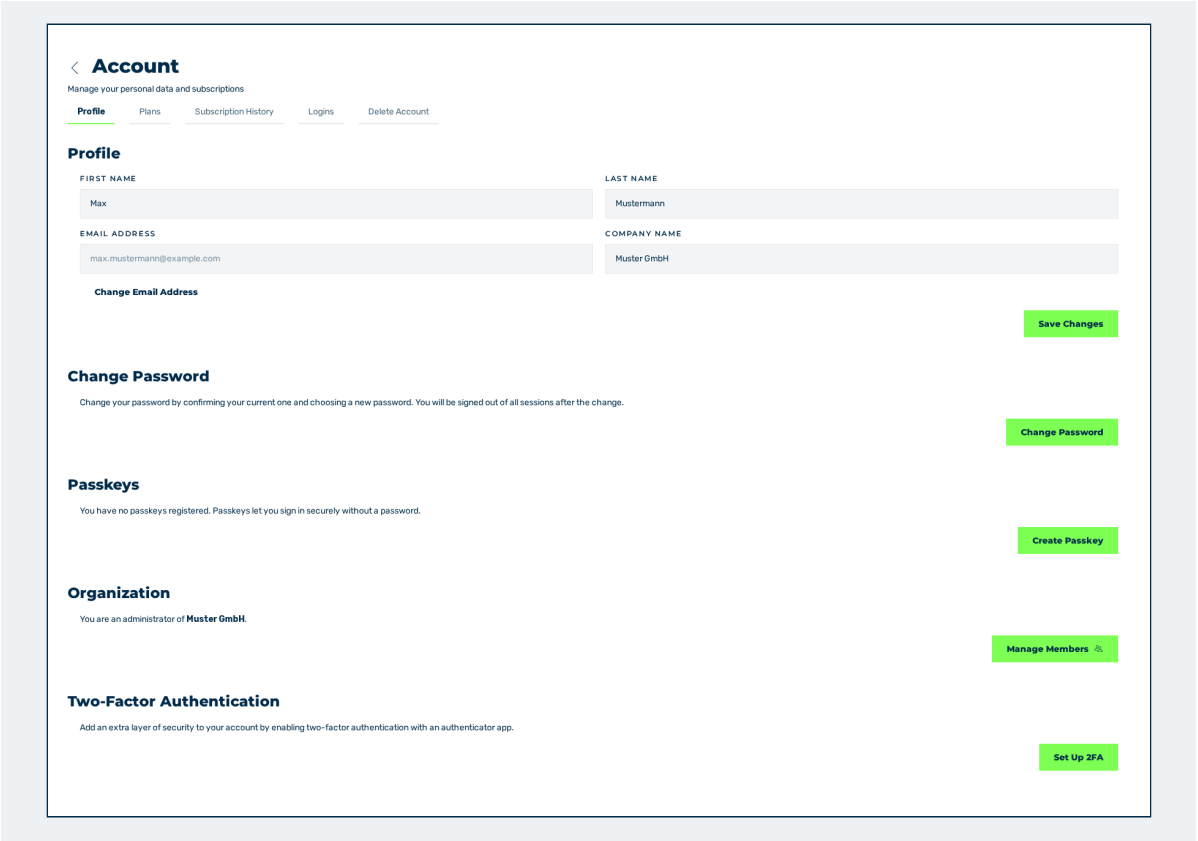


Abbildung 72: Reiter Profile

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile**.

► Klicken Sie auf die Schaltfläche **Change Email Address**.

↳ Der Dialog **Change Email Address** wird angezeigt. Der Dialog fordert zuerst Ihr aktuelles Passwort an.

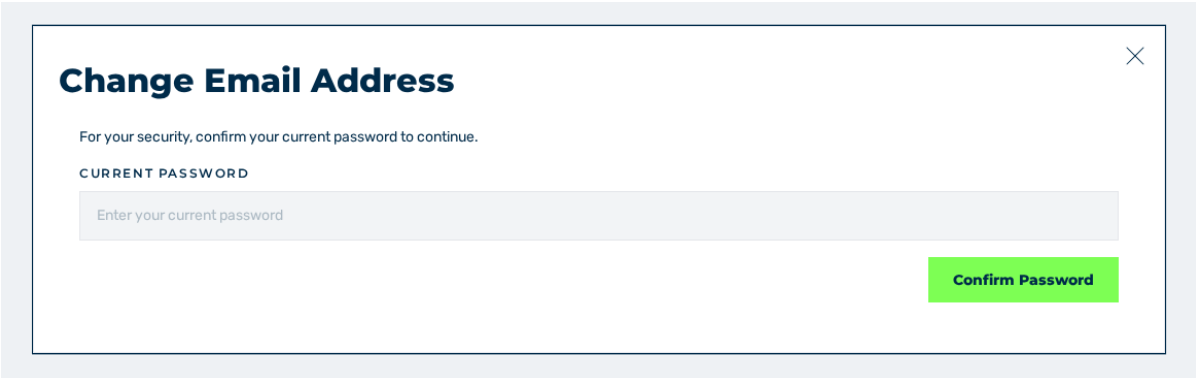


Abbildung 73: Dialog Change Email Address

- ▶ Geben Sie in das Feld **Current password** Ihr aktuelles Passwort ein.
- ▶ Klicken Sie auf die Schaltfläche **Confirm Password**.

  ↳ Der Dialog zeigt das Feld für die neue E-Mail-Adresse an. Der Text nennt das Postfach, an das die Bestätigung geht.

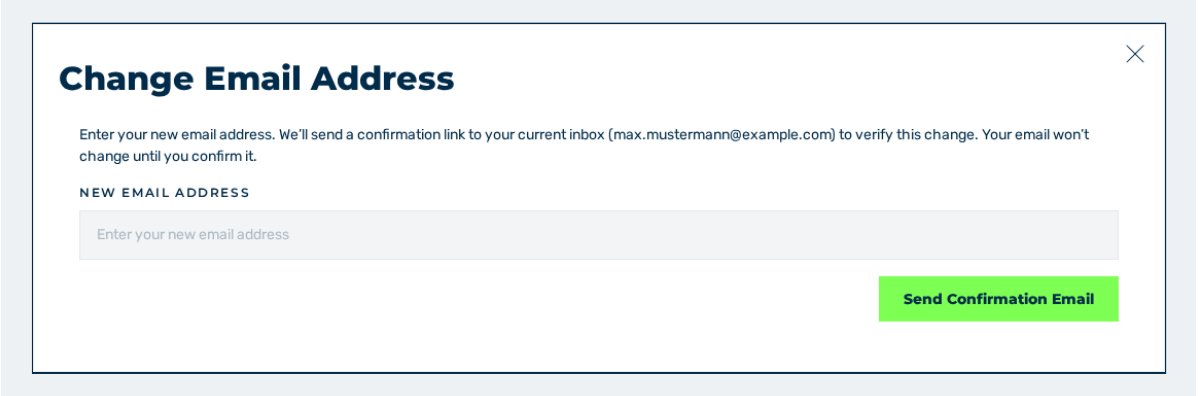


Abbildung 74: Dialog Change Email Address mit dem Feld New email address

- ▶ Geben Sie in das Feld **New email address** die neue E-Mail-Adresse ein.
- ▶ Klicken Sie auf die Schaltfläche **Send Confirmation Email**.

➔ Das System fordert die Änderung an und zeigt die Meldung **Email change requested** mit dem Text **Check your current inbox to confirm the change.** an.

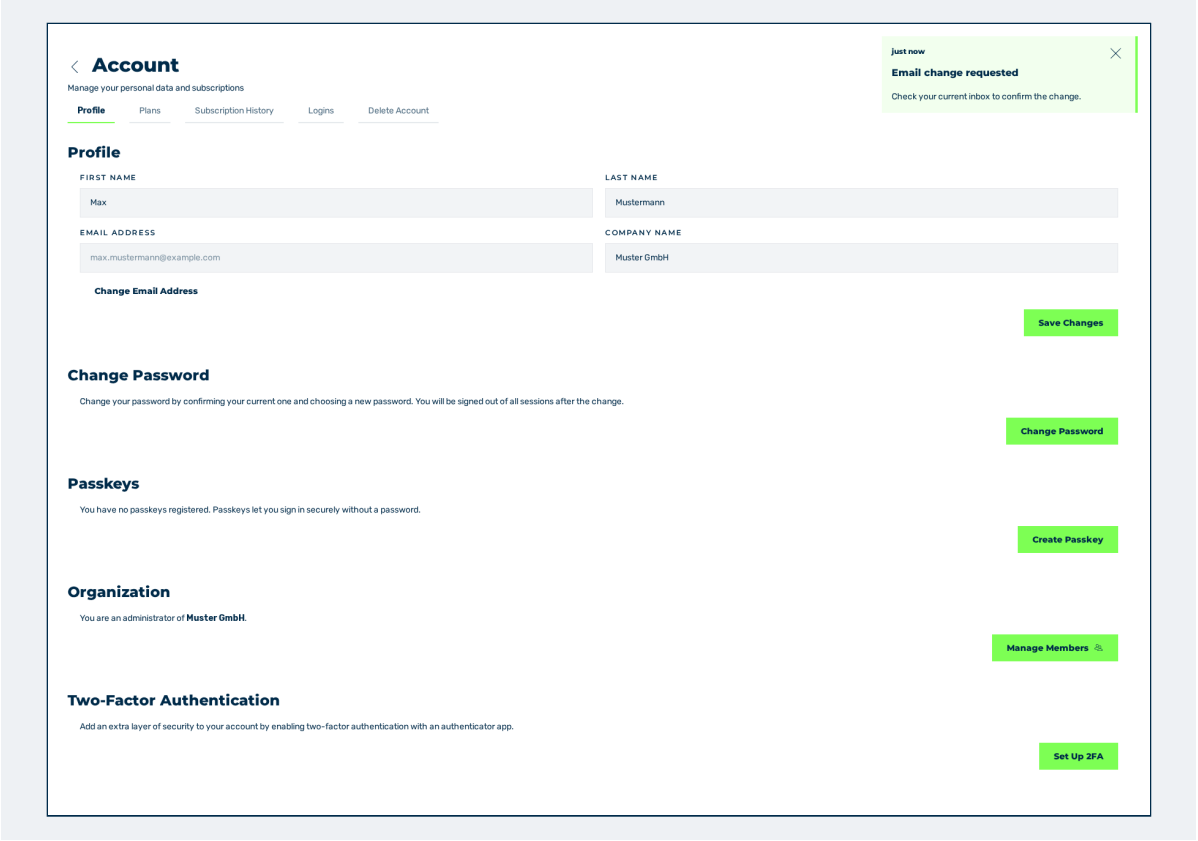


Abbildung 75: Reiter Profile mit der Meldung Email change requested

- ► Öffnen Sie das Postfach der bisherigen Adresse und bestätigen Sie die Änderung.
- ➔ Die neue E-Mail-Adresse ist wirksam.

Solange Sie die Änderung nicht bestätigt haben, bleibt die bisherige E-Mail-Adresse gültig.

4.3.3 Passwort ändern

Sie ändern das Passwort Ihres Kontos im Reiter **Profile**. Nach der Änderung meldet das Portal Sie ab.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet.

☑ Sie kennen Ihr aktuelles Passwort.

Um das Passwort zu ändern, gehen Sie wie folgt vor:

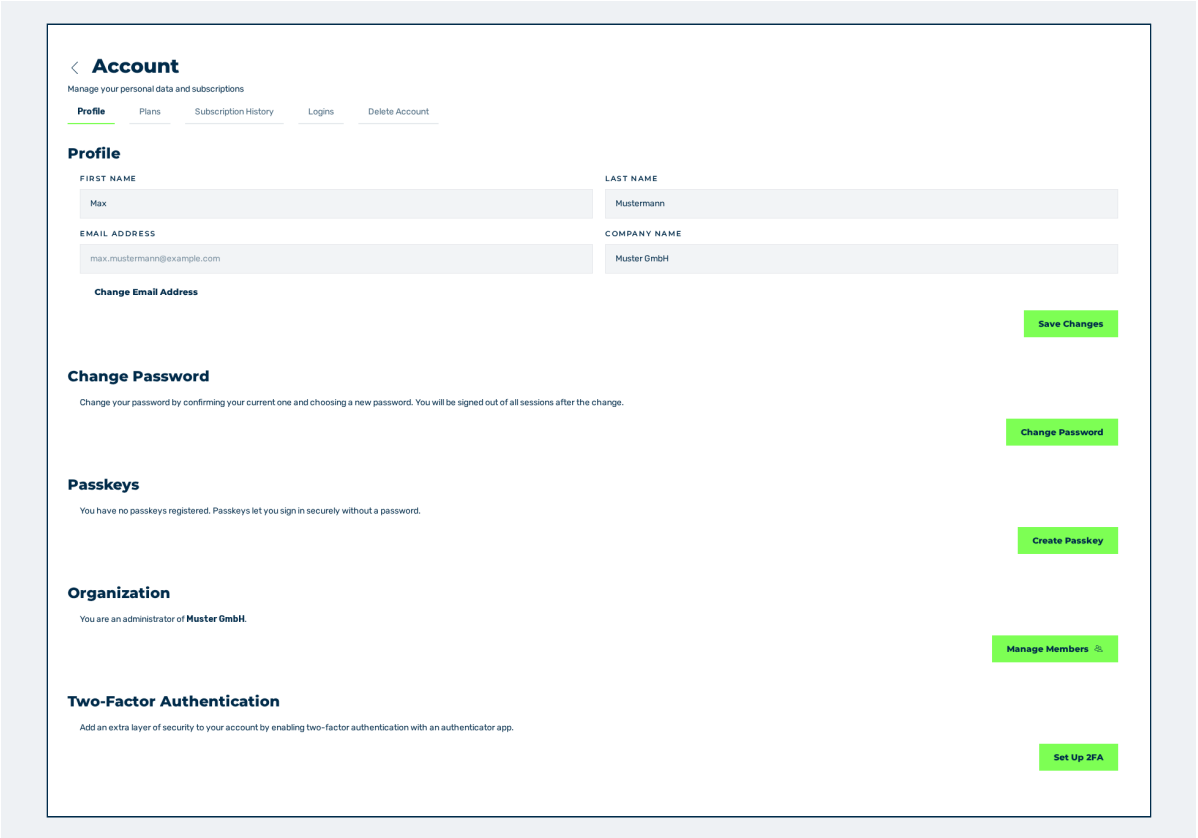


Abbildung 76: Reiter Profile

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile**.

► Klicken Sie auf die Schaltfläche **Change Password**.

↳ Der Dialog **Change Password** wird angezeigt.

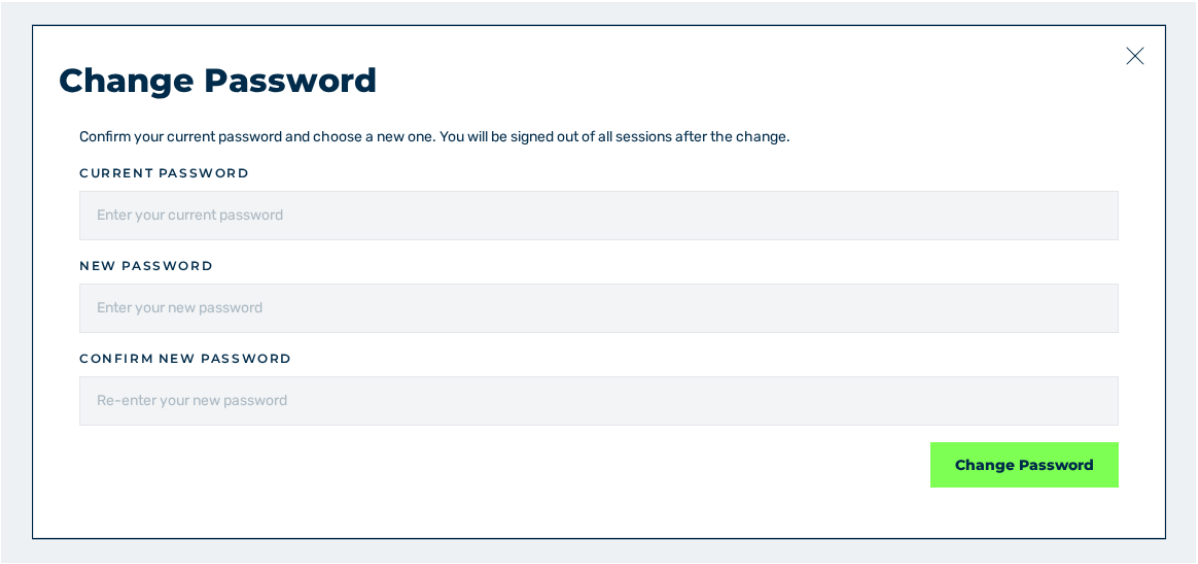


Abbildung 77: Dialog Change Password

- ▶ Geben Sie in das Feld **Current password** Ihr aktuelles Passwort ein.
- ▶ Geben Sie in das Feld **New password** das neue Passwort ein.
- ▶ Geben Sie in das Feld **Confirm new password** das neue Passwort erneut ein.
- ▶ Klicken Sie auf die Schaltfläche **Change Password**.
- ➔ Das System ändert das Passwort und zeigt die Meldung **Password changed** mit dem Text **Your password was changed. Please sign in again.** an. Die Anmeldeseite wird angezeigt.

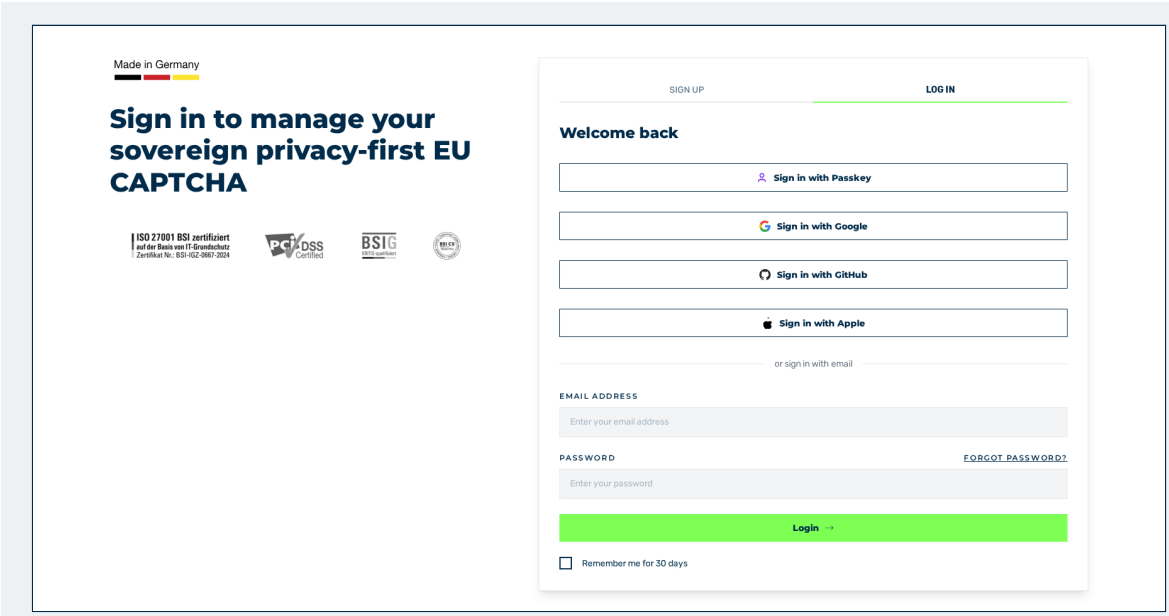


Abbildung 78: Ansicht Welcome back

► Melden Sie sich mit dem neuen Passwort an.

Siehe **Anmelden**.

4.3.4 Passwort zurücksetzen

Ist das Passwort verloren, fordern Sie über die Anmeldeseite eine E-Mail an und vergeben über den Link darin ein neues Passwort. Das Konto bleibt dabei erhalten.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie haben Zugriff auf das Postfach Ihrer E-Mail-Adresse.

Um das Passwort zurückzusetzen, gehen Sie wie folgt vor:

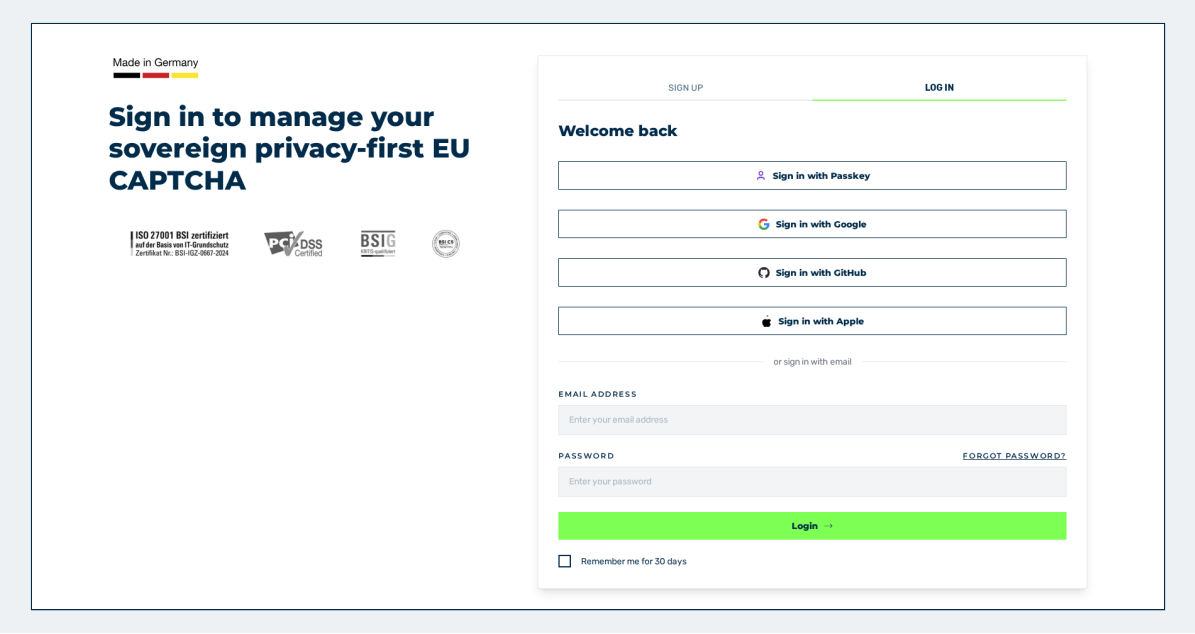


Abbildung 79: Ansicht Welcome back

- ▶ Rufen Sie die Anmeldeseite des Kundenportals auf.
- ▶ Klicken Sie auf den Link **Forgot Password?**.
  - ↳ Die Ansicht **Reset Password** öffnet sich.

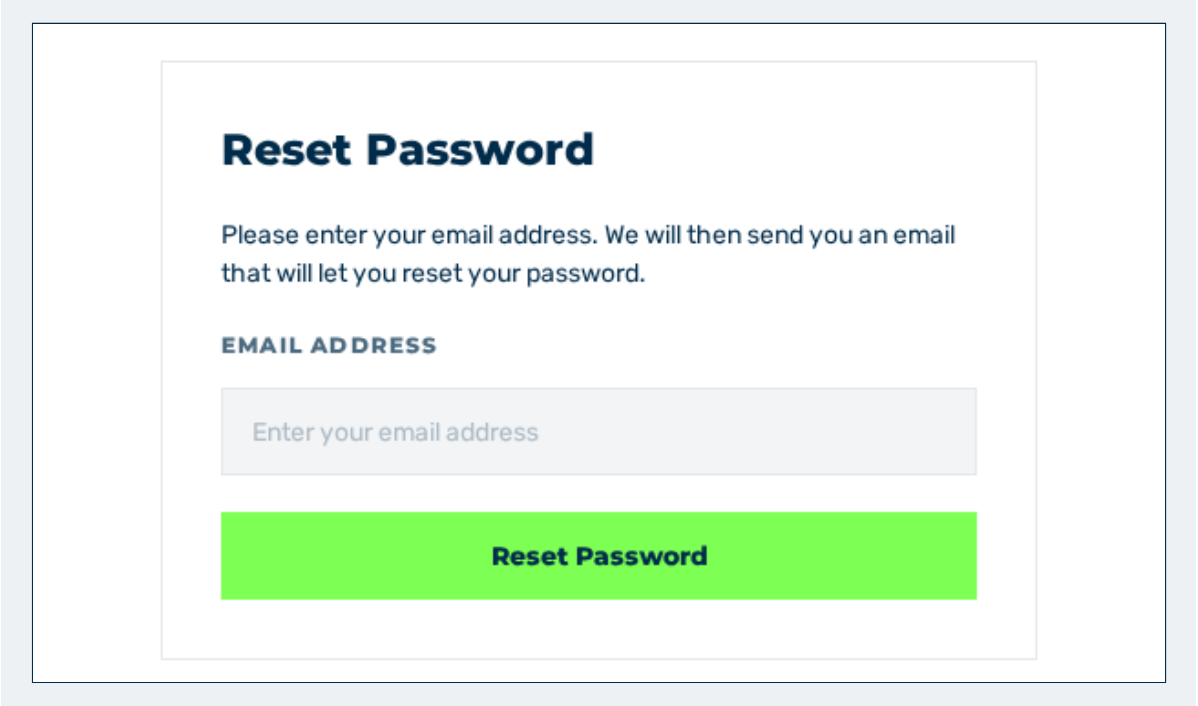


Abbildung 80: Ansicht Reset Password

- Geben Sie in das Feld **Email address** Ihre E-Mail-Adresse ein.
- Klicken Sie auf die Schaltfläche **Reset Password**.
  - ↳ Das System verschickt eine E-Mail mit einem Link an die angegebene Adresse.
- Folgen Sie dem Link in der E-Mail.
  - ↳ Die Ansicht **Reset Password** öffnet sich mit den Feldern für das neue Passwort.

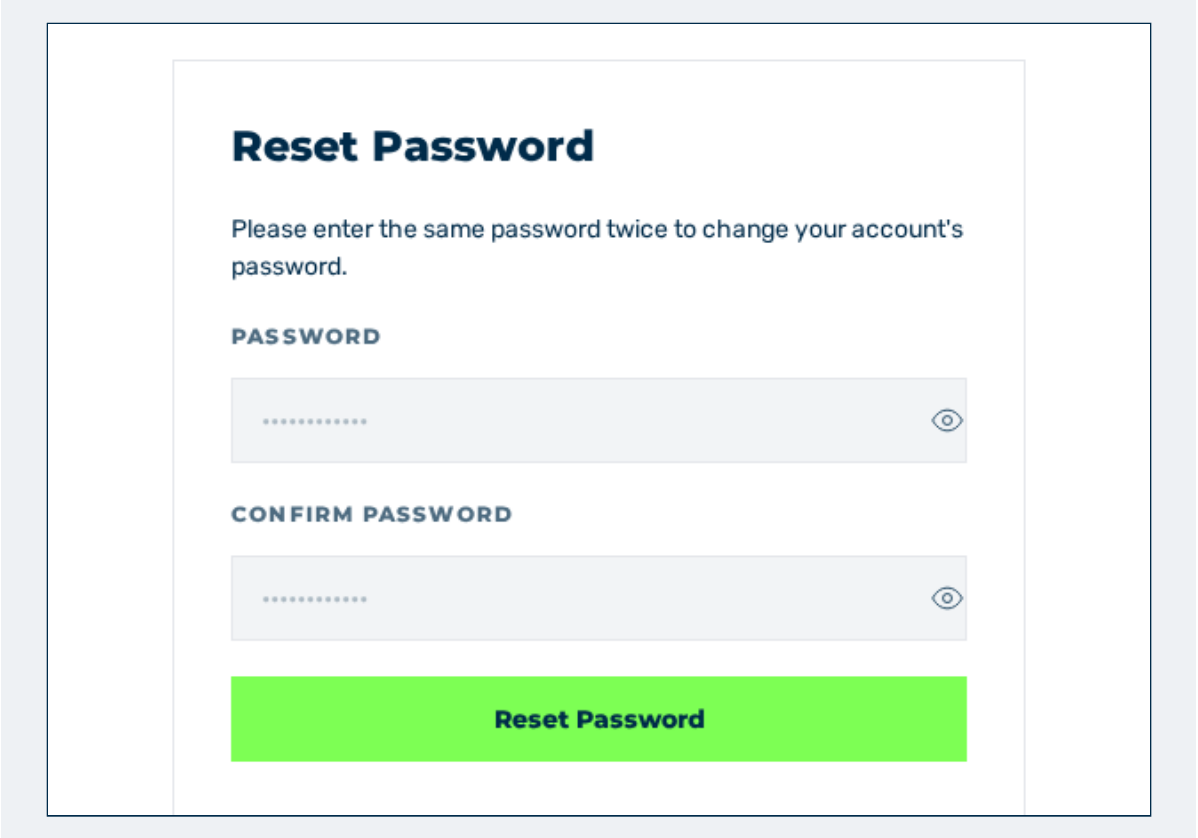


Abbildung 81: Ansicht Reset Password mit den Feldern Password und Confirm Password

- ▸ Geben Sie in das Feld **Password** das neue Passwort ein.
- ▸ Geben Sie in das Feld **Confirm Password** dasselbe Passwort ein.
- ▸ Klicken Sie auf die Schaltfläche **Reset Password**.
- ➔ Das System setzt das Passwort und zeigt die Anmeldeseite an. Melden Sie sich mit dem neuen Passwort an.

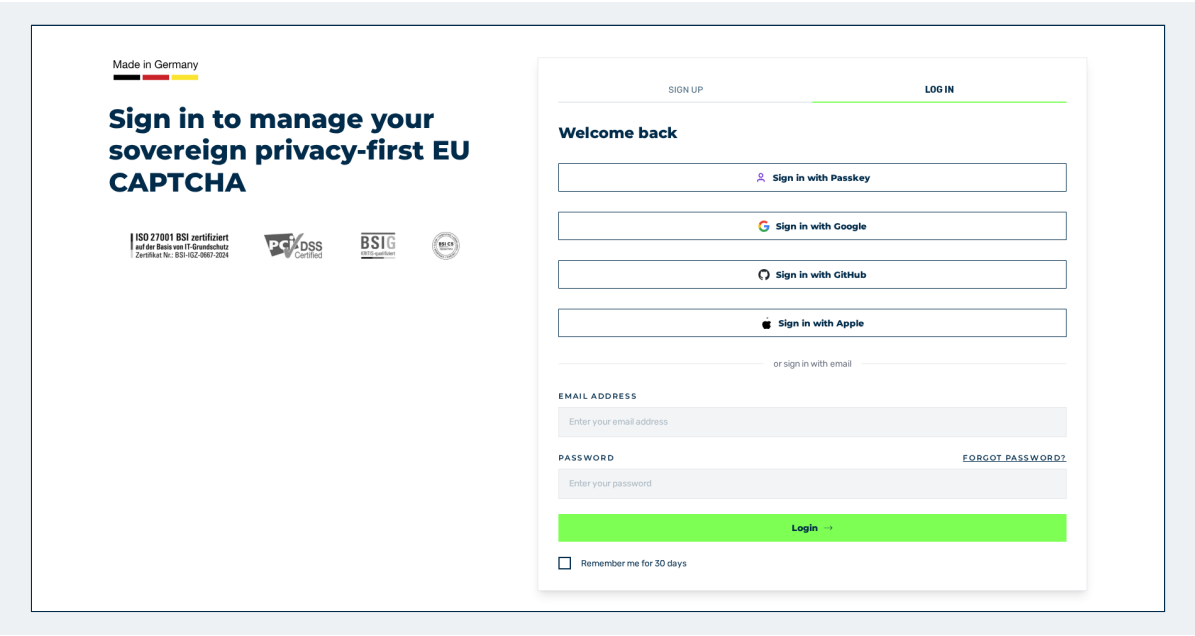


Abbildung 82: Ansicht Welcome back

4.3.4.1 Felder

Die folgenden Felder stehen zur Verfügung:

Feld	Inhalt
Email address	E-Mail-Adresse des Kontos, Platzhalter `Enter your email address`
Password	Neues Passwort. Eine Schaltfläche blendet die Eingabe ein und aus.
Confirm Password	Wiederholung des neuen Passworts

Nach dem Versand nennt das System den Hinweis, auch den Spam-Ordner zu prüfen. Die Schaltfläche **Back to Login** führt zurück zur Anmeldung.

4.3.4.2 Meldungen

Beim Zurücksetzen können die folgenden Meldungen auftreten:

Meldung	Ursache	Abhilfe
We couldn't find an account for that email address.	Zu der E-Mail-Adresse besteht kein Konto.	

Meldung	Ursache	Abhilfe
		Prüfen Sie die Adresse oder erstellen Sie ein Konto.
Passwords do not match, please enter the same password twice.	Die beiden Eingaben stimmen nicht überein.	Geben Sie in beiden Feldern dasselbe Passwort ein.
Your password reset link has expired. Please request a new one.	Der Link ist abgelaufen.	Fordern Sie eine neue E-Mail an.
This password reset link is invalid. Please request a new one.	Der Link ist ungültig.	Fordern Sie eine neue E-Mail an.
This password was used recently. Please choose a different one.	Das Passwort wurde für dieses Konto bereits verwendet.	Wählen Sie ein anderes Passwort.

Für das neue Passwort gelten dieselben Anforderungen wie bei der Registrierung. Siehe **Passwortrichtlinie**.

Siehe **Anmelden**.

4.3.5 Passwortrichtlinie

Für jedes Passwort gilt dieselbe Richtlinie — bei der Registrierung, beim Zurücksetzen und beim Ändern im Reiter **Profile**. Das System prüft jedes neue Passwort gegen die Richtlinie und lehnt es mit einer Meldung ab, sobald eine Regel verletzt ist.

4.3.5.1 Regeln

Ein Passwort muss die folgenden Regeln erfüllen:

Regel	Anforderung
Länge	Mindestens 12 Zeichen
Großbuchstaben	Mindestens ein Großbuchstabe
Kleinbuchstaben	Mindestens ein Kleinbuchstabe
Ziffern	Mindestens eine Ziffer
Sonderzeichen	Mindestens ein Sonderzeichen

Als Sonderzeichen gelten die folgenden Zeichen:

```
#()^,.<>/\|{}[]-=_+$@!%*?&
```

Die Mindestlänge ist serverseitig über die Umgebungsvariable `PASSWORD_MIN_LENGTH` einstellbar. Ohne gesetzte Variable gilt der Wert 12.

4.3.5.2 Prüfung während der Eingabe

Sobald Sie ein Passwortfeld ausfüllen, steht unter dem Feld die Liste der Regeln. Erfüllte Regeln sind markiert:

- **At least [Anzahl] characters**
- **At least one uppercase character**
- **At least one lowercase character**
- **At least one number**
- **At least one special character ([Zeichen])**

Eine Schaltfläche am rechten Rand des Feldes blendet das Passwort ein und aus. Der Kurzhinweis wechselt zwischen **Show password** und **Hide password**.

4.3.5.3 Bestätigung und Wiederverwendung

Zusätzlich zu den Regeln gilt:

- Die Bestätigung muss mit dem neuen Passwort übereinstimmen.
- Ein kürzlich verwendetes Passwort wird abgelehnt. Beim Zurücksetzen und beim Ändern vergleicht das System das neue Passwort mit den zuletzt verwendeten Passwörtern des Kontos. Bei der Registrierung entfällt die Prüfung, weil noch kein früheres Passwort besteht.

4.3.5.4 Meldungen

Die folgenden Meldungen können auftreten:

Meldung	Ursache
Your password does not meet the requirements.	Das Passwort verletzt eine Regel zu Länge oder Zusammensetzung.
The passwords do not match.	Das neue Passwort und die Bestätigung stimmen nicht überein.
Passwords do not match.	Die Bestätigung weicht vom neuen Passwort ab. Die Meldung steht direkt unter dem Bestätigungsfeld, noch vor dem Absenden.
This password was used recently. Please choose a different one.	Das Passwort wurde für dieses Konto bereits verwendet.
The current password is incorrect.	Beim Ändern wurde das bisherige Passwort falsch eingegeben.
Too many attempts. Please try again later.	Zu viele Versuche in kurzer Folge.
Password login is not enabled for this account.	Das Konto meldet sich über einen Anbieter an.
	Der Link zum Zurücksetzen ist abgelaufen.

Meldung	Ursache
Your password reset link has expired. Please request a new one.	
This password reset link is invalid. Please request a new one.	Der Link zum Zurücksetzen ist ungültig.
We couldn't find an account for that email address.	Zu der E-Mail-Adresse besteht kein Konto.

Ein Passwort allein schützt das Konto nicht ausreichend. Richten Sie zusätzlich einen Passkey oder die Zwei-Faktor-Authentifizierung ein.

Siehe **Passwort ändern**.

Siehe **Passwort zurücksetzen**.

4.3.6 Passkey erstellen

Ein Passkey ermöglicht die Anmeldung ohne Passwort. Das Gerät bestätigt die Anmeldung, zum Beispiel über Fingerabdruck, Gesichtserkennung oder Geräte-PIN.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet.

☑ Ihr Gerät und Ihr Browser unterstützen Passkeys.

Um einen Passkey zu erstellen, gehen Sie wie folgt vor:

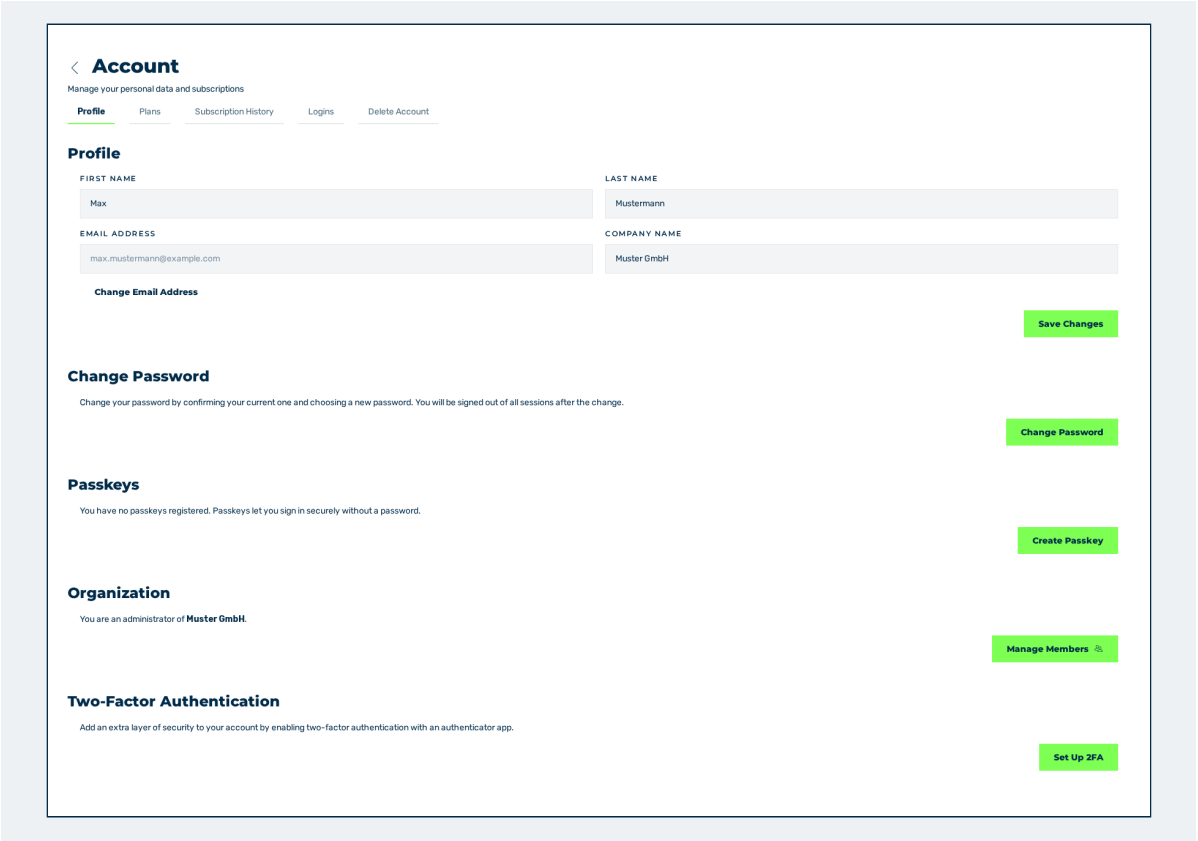


Abbildung 83: Reiter Profile

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile**.

► Klicken Sie auf die Schaltfläche **Create Passkey**.

↳ Der Browser fordert die Bestätigung durch das Gerät an.

► Bestätigen Sie die Abfrage auf Ihrem Gerät.

➔ Das System legt den Passkey an und zeigt die Meldung **Passkey registered successfully!** an.

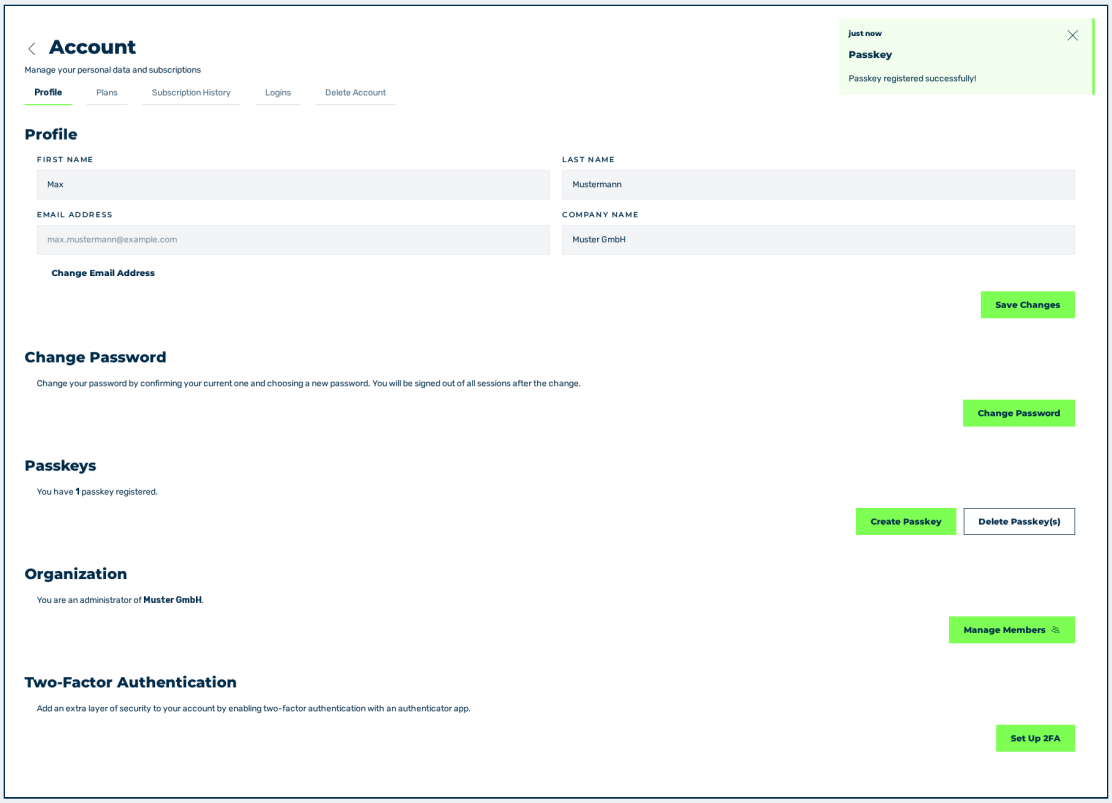


Abbildung 84: Reiter Profile mit angelegtem Passkey

Um alle Passkeys zu entfernen, klicken Sie auf die Schaltfläche **Delete Passkey(s)**. Siehe **Passkeys löschen**.

Die Anmeldung über Passkey ist derzeit gestört. Verwenden Sie bis zur Behebung die Anmeldung mit E-Mail-Adresse und Passwort. Siehe **Anmeldung über Passkey schlägt fehl**.

4.3.7 Passkeys löschen

Die Schaltfläche **Delete Passkey(s)** entfernt alle Passkeys des Kontos auf einmal. Einzelne Passkeys lassen sich nicht löschen.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet.

☑ Für Ihr Konto ist mindestens ein Passkey angelegt.

Um alle Passkeys zu löschen, gehen Sie wie folgt vor:

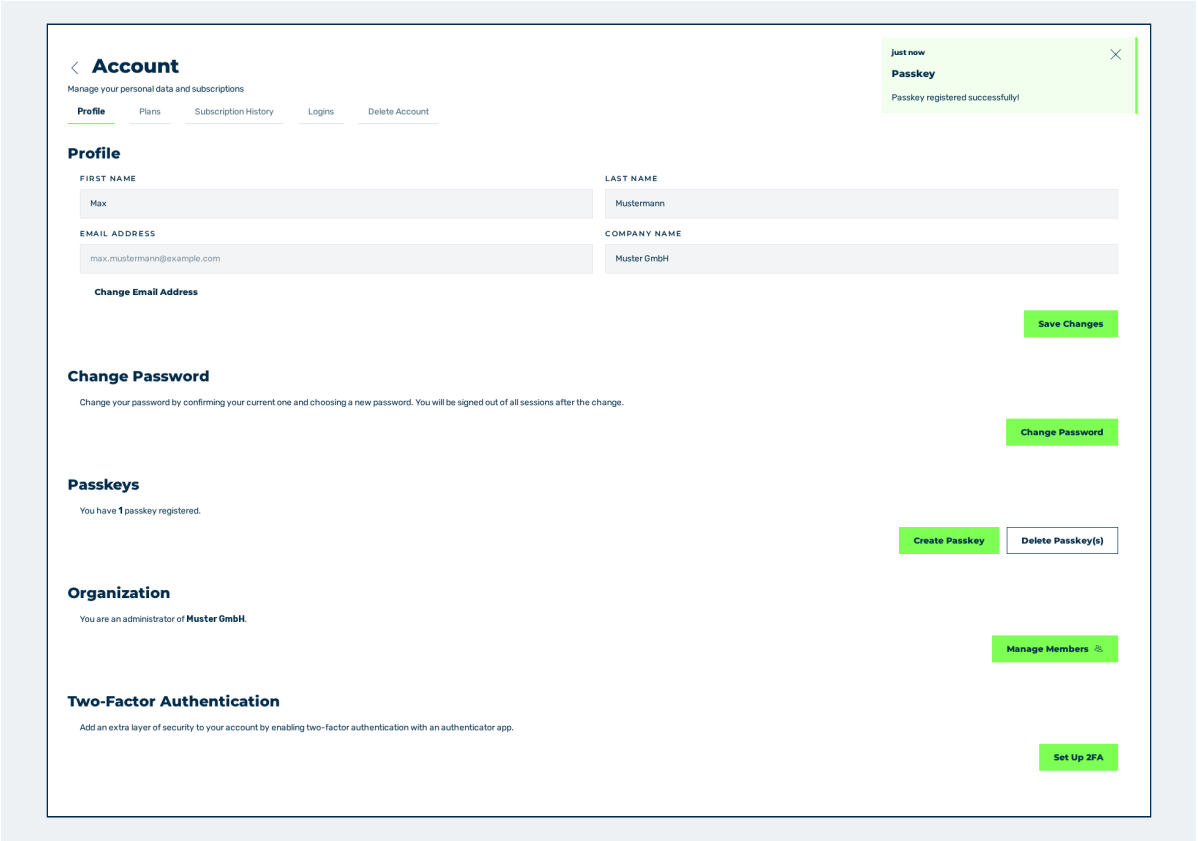


Abbildung 85: Reiter Profile mit einem angelegten Passkey

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile**.

► Klicken Sie im Bereich **Passkeys** auf die Schaltfläche **Delete Passkey(s)**.

➔ Das System löscht alle Passkeys und zeigt die Meldung **All passkeys have been deleted.** an.

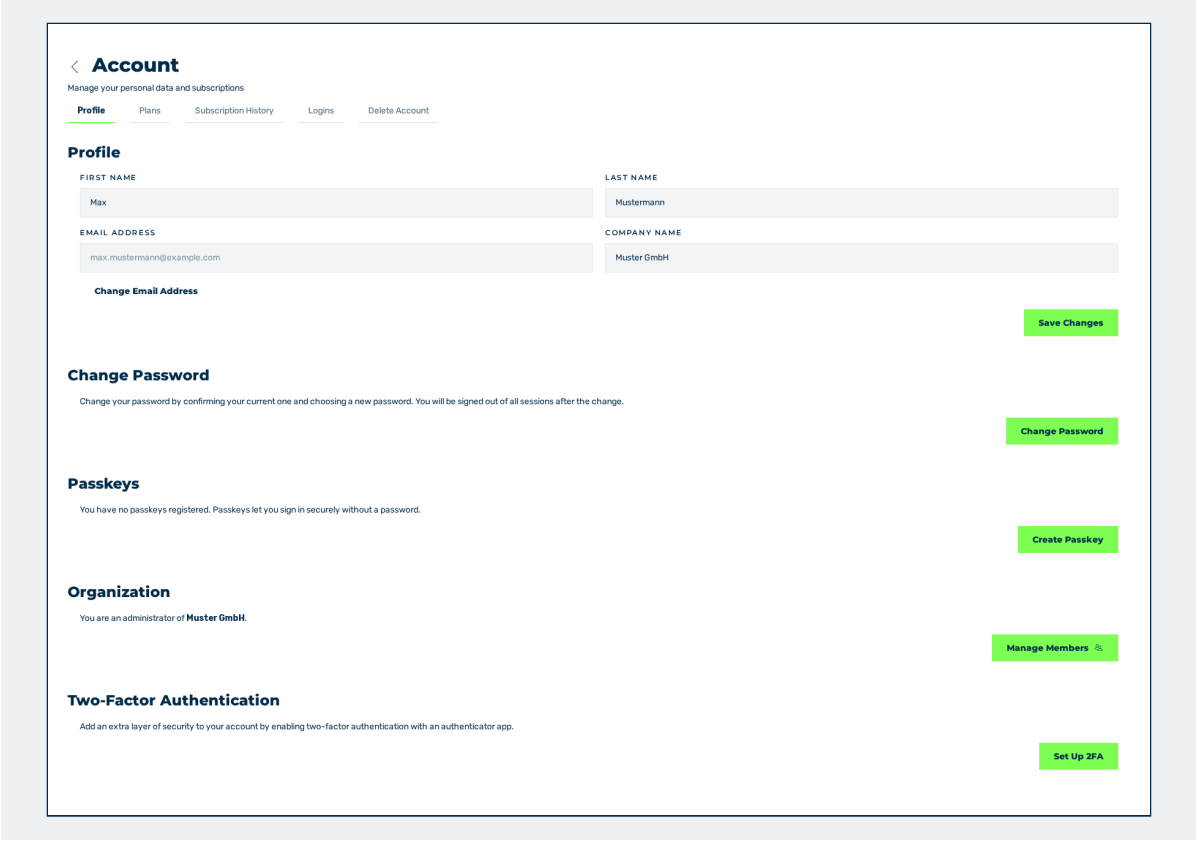


Abbildung 86: Reiter Profile ohne angelegte Passkeys

Die Schaltfläche **Delete Passkey(s)** erscheint erst mit dem ersten angelegten Passkey.

> Nach dem Löschen ist die Anmeldung über einen Passkey nicht mehr möglich. Melden Sie sich mit E-Mail-Adresse und Passwort oder über einen Anbieter an.

Siehe **Passkey erstellen**.

4.3.8 Zwei-Faktor-Authentifizierung einrichten

Die Zwei-Faktor-Authentifizierung verlangt bei jeder Anmeldung zusätzlich einen sechsstelligen Code aus einer Authenticator-App.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet.

☑ Auf Ihrem Mobilgerät ist eine Authenticator-App installiert.

Um die Zwei-Faktor-Authentifizierung einzurichten, gehen Sie wie folgt vor:

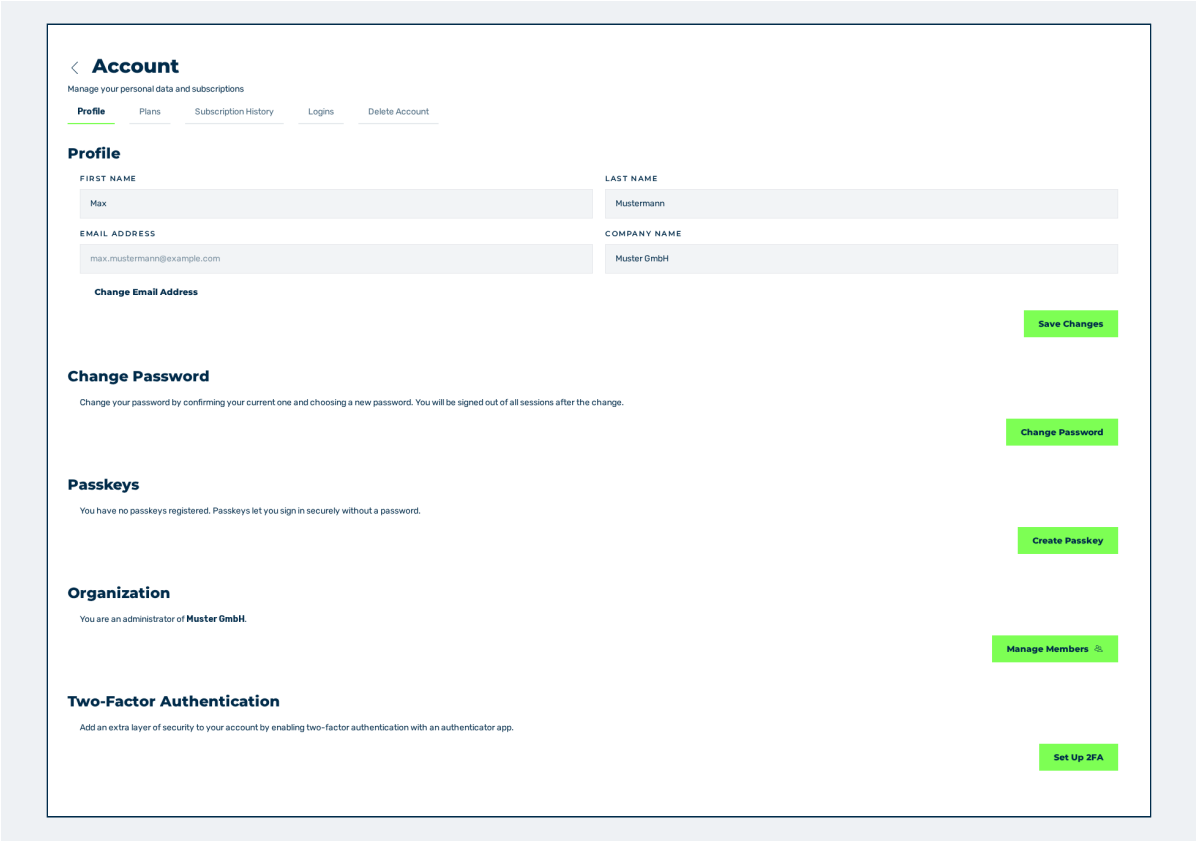


Abbildung 87: Reiter Profile

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile**.

► Klicken Sie auf die Schaltfläche **Set Up 2FA**.

↳ Die Ansicht **Set Up Two-Factor Authentication** wird angezeigt.

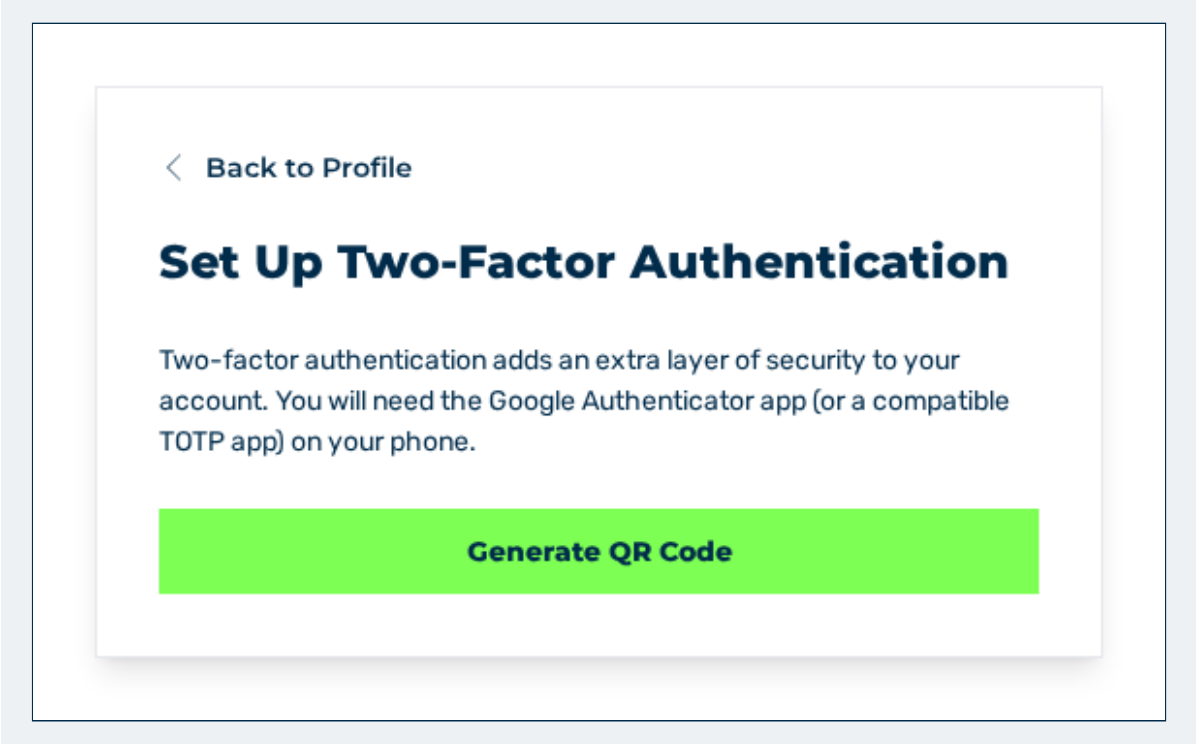


Abbildung 88: Ansicht Set Up Two-Factor Authentication

► Klicken Sie auf die Schaltfläche **Generate QR Code**.

↳ Das System zeigt den QR-Code an. Darunter steht unter **Or enter this secret manually:** das Geheimnis in Textform, zum Beispiel `XXXX XXXX XXXX XXXX`.

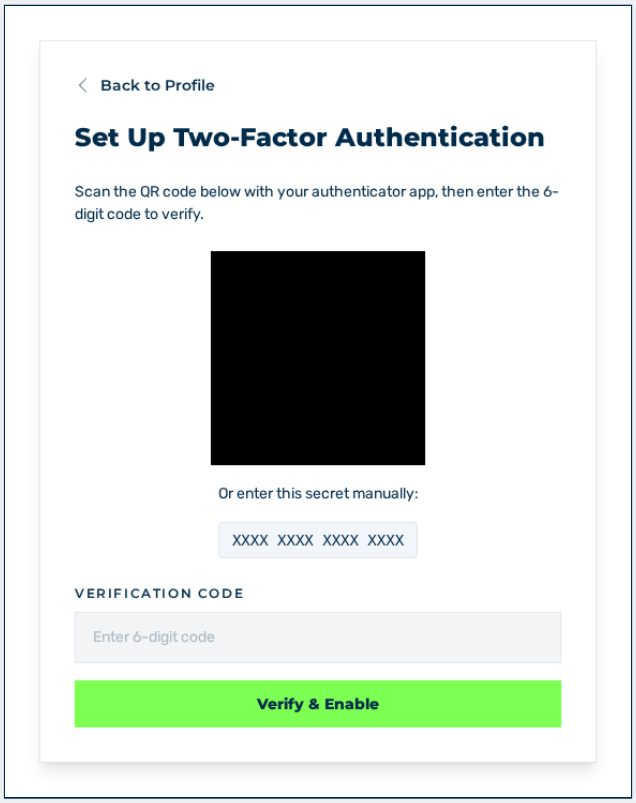


Abbildung 89: Ansicht Set Up Two-Factor Authentication mit maskiertem QR-Code

- ► Scannen Sie den QR-Code mit Ihrer Authenticator-App. Alternativ geben Sie das Geheimnis in der App ein.
- ► Geben Sie in das Feld **Verification code** den sechsstelligen Code aus der App ein.
- ► Klicken Sie auf die Schaltfläche **Verify & Enable**.
- ➔ Das System aktiviert die Zwei-Faktor-Authentifizierung und zeigt die Meldung **Two-factor authentication has been enabled successfully.** an.

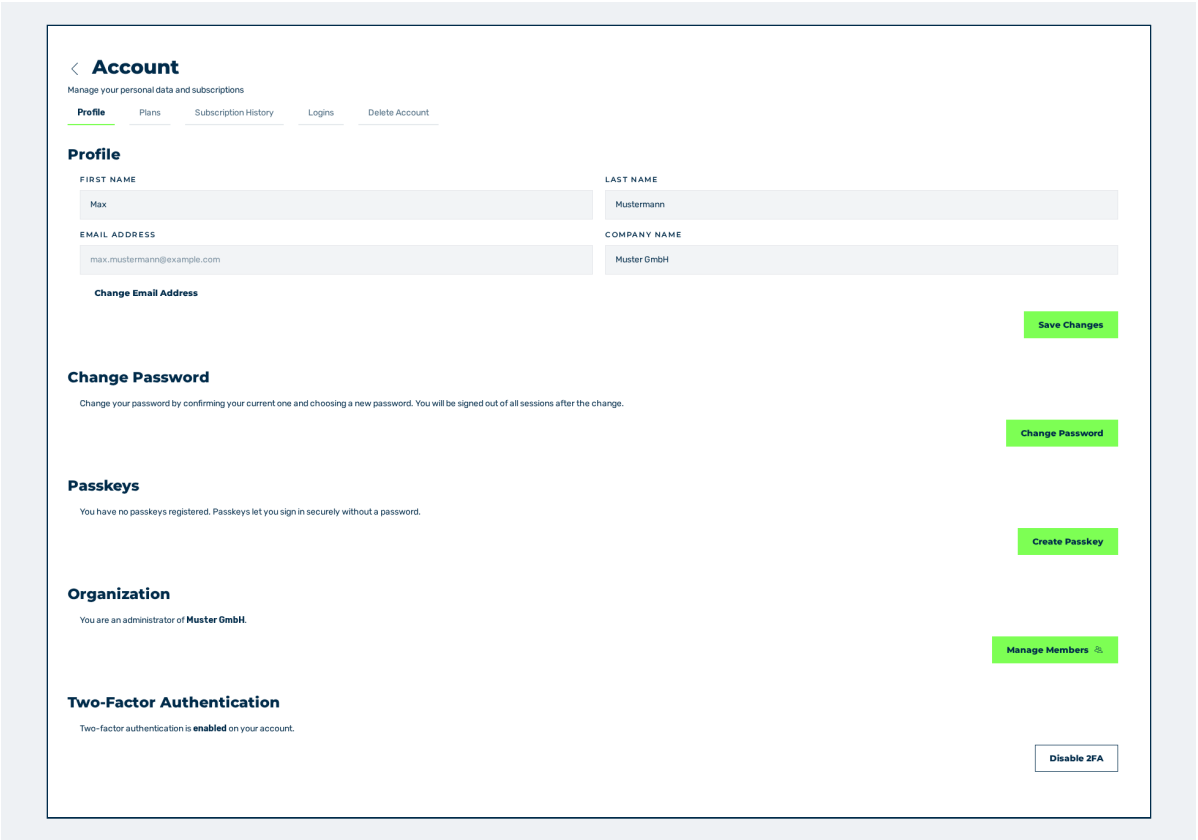


Abbildung 90: Reiter Profile mit aktivierter Zwei-Faktor-Authentifizierung

► Klicken Sie auf die Schaltfläche **Back to Profile**.

Bewahren Sie das Geheimnis sicher auf. Ohne Authenticator-App und ohne Geheimnis ist die Anmeldung nicht mehr möglich.

Siehe **Zwei-Faktor-Authentifizierung deaktivieren**.

4.3.9 Zwei-Faktor-Authentifizierung deaktivieren

Nach der Deaktivierung genügen E-Mail-Adresse und Passwort für die Anmeldung.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet.

☑ Für Ihr Konto ist die Zwei-Faktor-Authentifizierung aktiv.

Um die Zwei-Faktor-Authentifizierung zu deaktivieren, gehen Sie wie folgt vor:

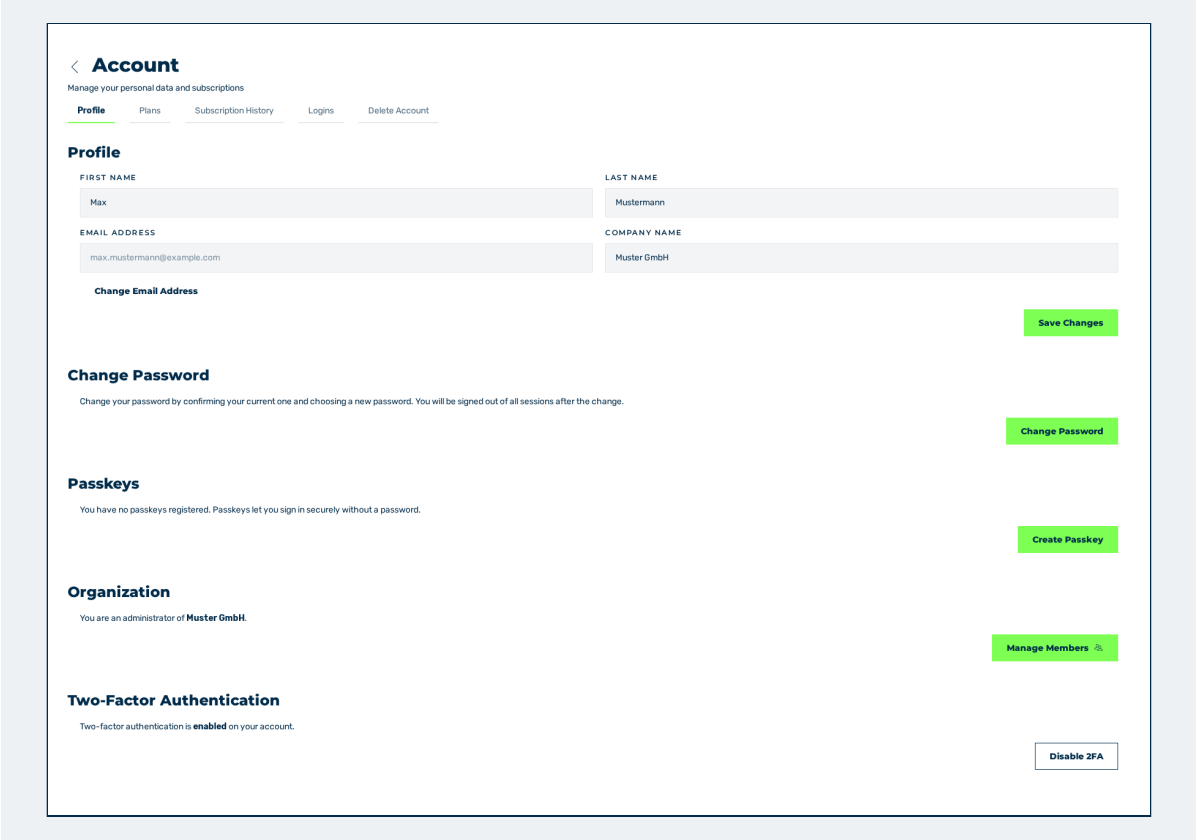


Abbildung 91: Reiter Profile

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile**.

► Klicken Sie auf die Schaltfläche **Disable 2FA**.

↳ Der Dialog **Disable Two-Factor Authentication** wird angezeigt.

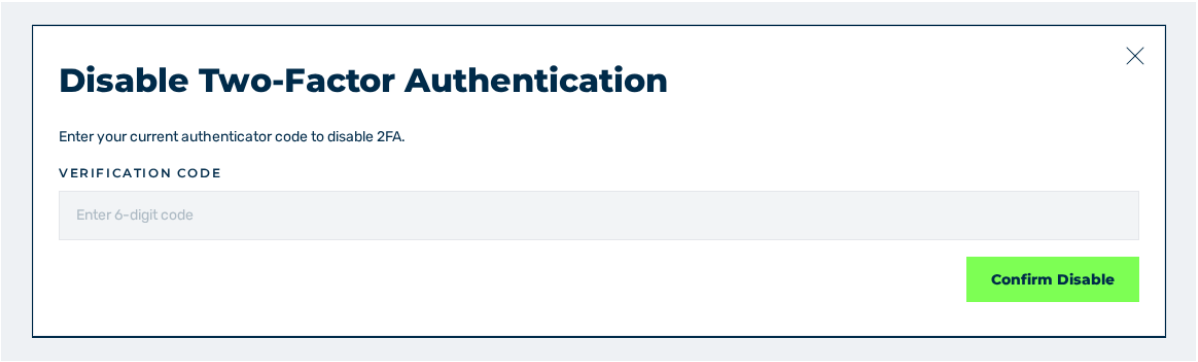


Abbildung 92: Dialog Disable Two-Factor Authentication

- ▶ Geben Sie in das Feld **Verification code** den sechsstelligen Code aus Ihrer Authenticator-App ein.
- ▶ Klicken Sie auf die Schaltfläche **Confirm Disable**.
- ➔ Das System deaktiviert die Zwei-Faktor-Authentifizierung und zeigt die Meldung **Two-factor authentication has been disabled.** an.

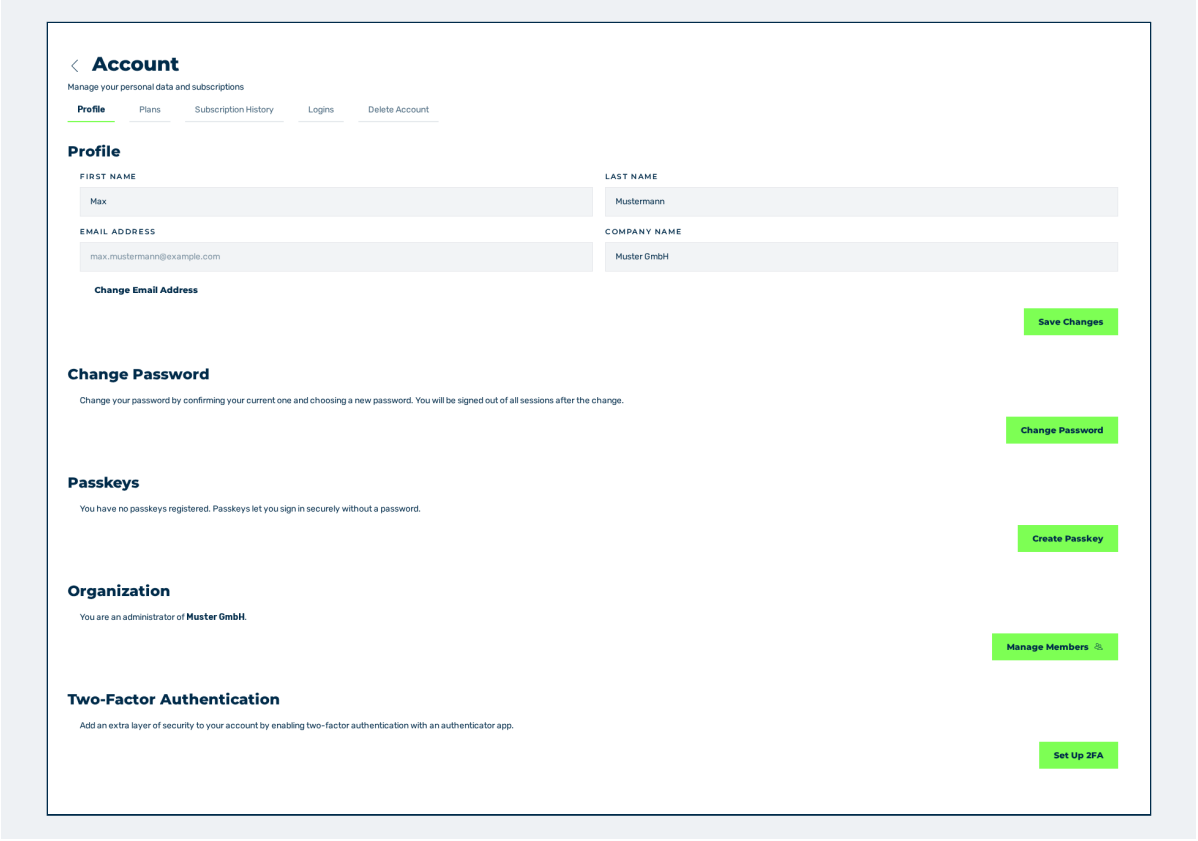


Abbildung 93: Reiter Profile ohne Zwei-Faktor-Authentifizierung

Ohne Zwei-Faktor-Authentifizierung schützt allein das Passwort Ihr Konto. Prüfen Sie regelmäßig das Protokoll der Anmeldungen. Siehe **Reiter Logins**.

4.3.10 Konto löschen

Der Reiter **Delete Account** löscht Ihr Konto endgültig. Mit dem Konto verschwinden alle Sitekeys, damit auch der Schutz der zugehörigen Websites.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet.

☑ Für Ihr Konto ist kein Abonnement mehr aktiv. Siehe **Abonnement kündigen**.

☑ Für Ihr Konto wird keine Zahlung mehr wiederholt.

Um das Konto zu löschen, gehen Sie wie folgt vor:

Abbildung 94: Reiter Delete Account

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile**.

► Wechseln Sie in den Reiter **Delete Account**.

► Klicken Sie auf die Schaltfläche **Delete Account Permanently**.

↳ Der Dialog **Delete Account Permanently** öffnet sich.

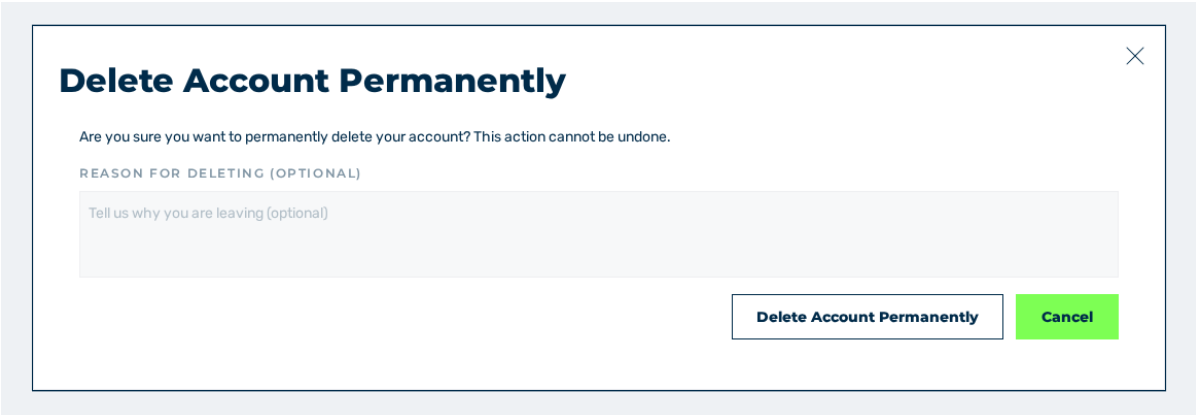


Abbildung 95: Dialog Delete Account Permanently

- ► Falls erforderlich, geben Sie in das Feld **Reason for deleting (optional)** eine Begründung ein.
- ► Klicken Sie auf die Schaltfläche **Delete Account Permanently**.
- ➔ Das System löscht das Konto und meldet Sie ab.

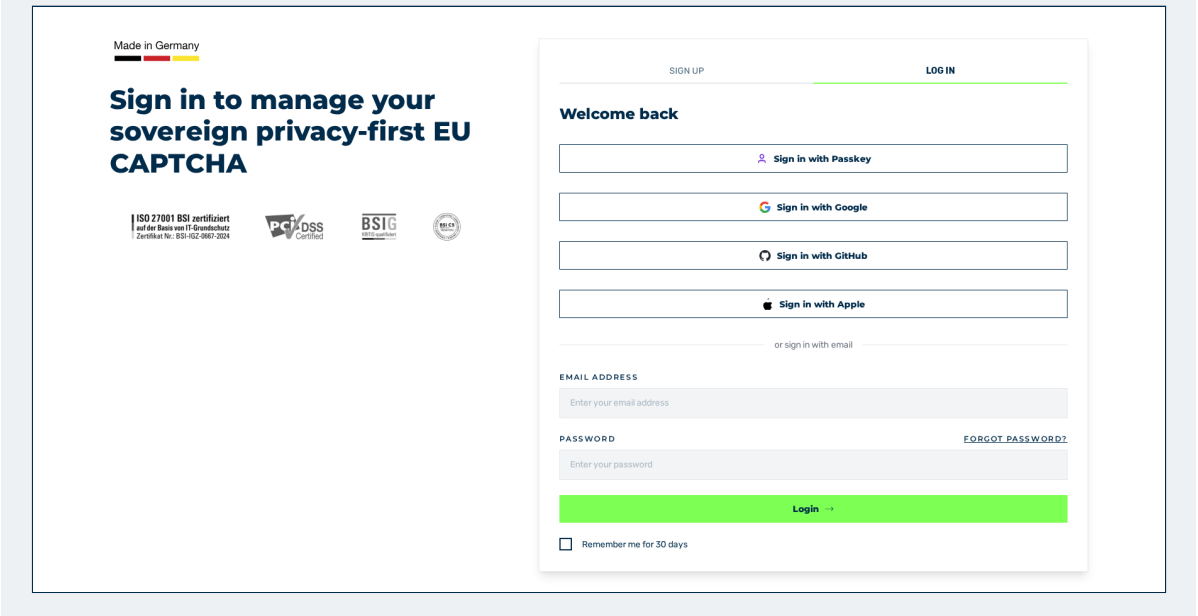


Abbildung 96: Ansicht Welcome back nach der Löschung

Die Begründung ist freiwillig und fasst höchstens 2048 Zeichen. Um den Vorgang abzubrechen, klicken Sie im Dialog auf die Schaltfläche **Cancel**.

Die Löschung lässt sich nicht rückgängig machen. Ihr Konto und Ihre persönlichen Daten werden endgültig gelöscht, alle Sitekeys und die zugehörigen CAPTCHA-Daten werden entfernt, und eine erneute Anmeldung ist nicht möglich.

4.3.10.1 Blockierte Löschung

Solange eine der folgenden Bedingungen zutrifft, lehnt das System die Löschung ab:

Bedingung	Titel des Dialogs	Maßnahme
Ein Abonnement ist aktiv.	**Cancel your subscription first**	Kündigen Sie das Abonnement. Die Schaltfläche **Go to Subscription History** öffnet den zugehörigen Reiter.
Eine Zahlung wird noch wiederholt.	**A payment is still being retried**	Warten Sie den Abschluss der Zahlung ab.

Die Schaltfläche **Close** schließt den Dialog.

Siehe **Reiter Delete Account**.

4.3.11 Cookie- und Tracking-Einstellungen

Das Kundenportal wertet die Nutzung der Oberfläche mit Analysewerkzeugen aus. Beim ersten Aufruf fragt der Dialog **We use tracking tools to improve your experience** die Einwilligung ab. Die Einwilligung ist freiwillig und jederzeit widerrufbar.

> Die Einstellungen betreffen ausschließlich das Kundenportal. Der Schutz selbst arbeitet ohne Cookies: Das Widget setzt auf der geschützten Website keine Cookies und speichert nichts im Browser der Besucher. Siehe Funktionsweise.

4.3.11.1 Kategorien

Die Einwilligung gliedert sich in die folgenden Kategorien:

Kategorie	Inhalt	Vorgabe
Necessary	Grundfunktionen, Sicherheit und Funktionsfähigkeit der Website. Nicht abwählbar.	**Enabled**
User Analytics	Auswertung der Nutzung von Website und Produkt über Matomo und Userpilot.	**Disabled**
Session Replay	Aufzeichnung pseudonymisierter Sitzungsdaten wie Mausbewegungen, Klicks und Scrollverhalten über Userpilot.	**Disabled**

> **Session Replay** setzt **User Analytics** voraus. Schalten Sie **Session Replay** ein, wird **User Analytics** mit eingeschaltet; schalten Sie **User Analytics** aus, wird **Session Replay** mit ausgeschaltet.

4.3.11.2 Einwilligung erteilen

Um die Einwilligung im ersten Dialog zu erteilen, gehen Sie wie folgt vor:

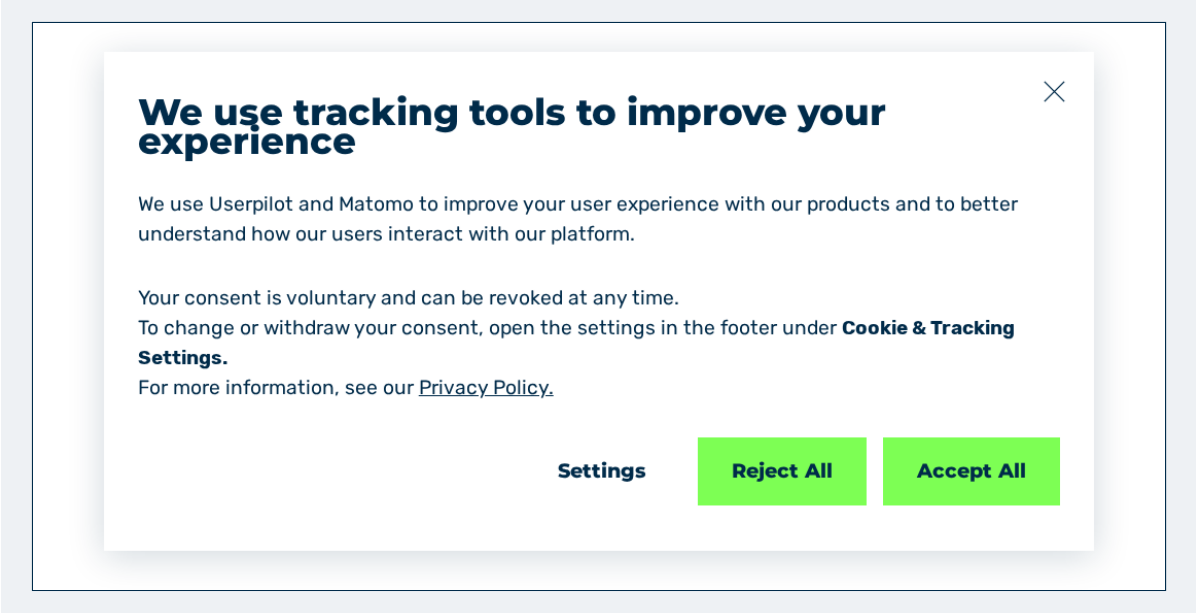


Abbildung 97: Dialog We use tracking tools to improve your experience

► Klicken Sie auf eine der folgenden Schaltflächen:

Schaltfläche	Wirkung
Accept All	Schaltet alle Kategorien ein.
Reject All	Schaltet **User Analytics** und **Session Replay** aus.
Settings	Zeigt die einzelnen Kategorien zur Auswahl an.

➔ Das System speichert die Auswahl und schließt den Dialog.

4.3.11.3 Kategorien einzeln wählen

Um einzelne Kategorien zu wählen, gehen Sie wie folgt vor:

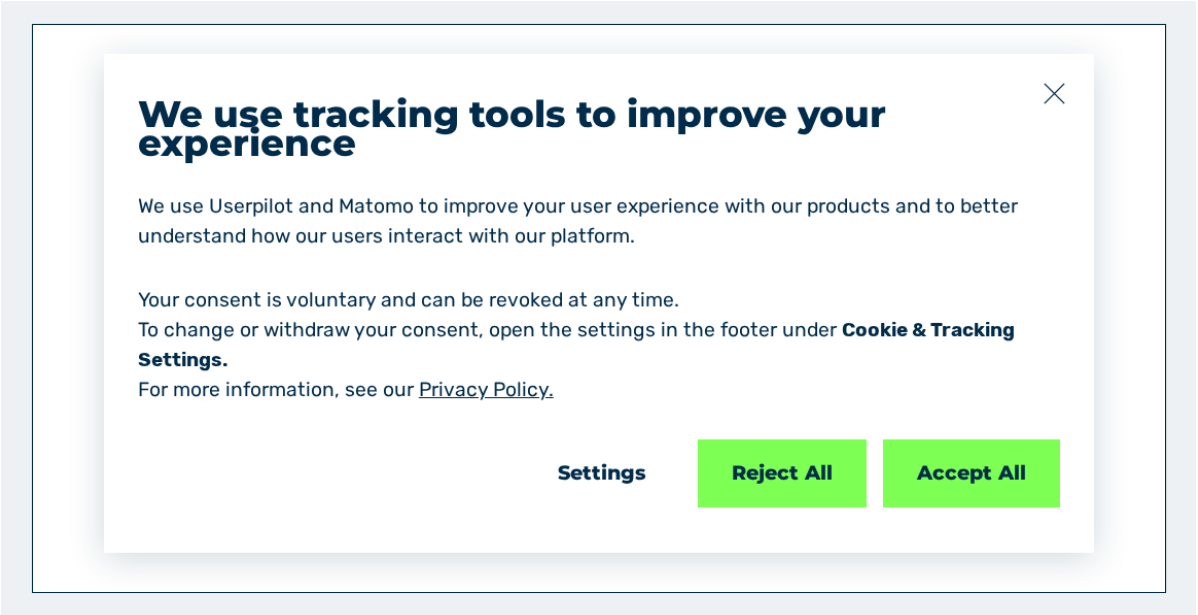


Abbildung 98: Dialog mit den Kategorien

- ► Klicken Sie im Dialog auf die Schaltfläche **Settings**.
  - ↳ Der Dialog zeigt die drei Kategorien mit je einem Schalter an.

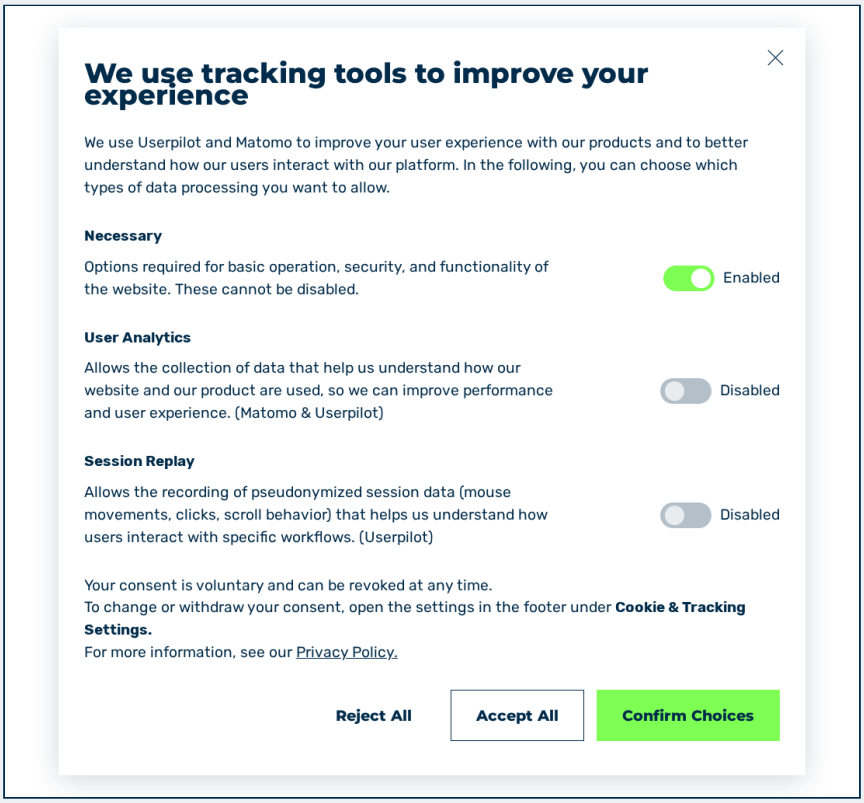


Abbildung 99: Dialog mit den Schaltern der drei Kategorien

- ► Stellen Sie die gewünschten Kategorien auf **Enabled**.
- ► Klicken Sie auf die Schaltfläche **Confirm Choices**.
- ➔ Das System verwendet nur noch die gewählten Kategorien und schließt den Dialog.

4.3.11.4 Einwilligung ändern oder widerrufen

Um die Einwilligung nachträglich zu ändern, gehen Sie wie folgt vor:

- ► Klicken Sie in der Fußzeile auf den Eintrag **Cookie & Tracking Settings**.
  - ↳ Der Dialog mit den Kategorien öffnet sich.

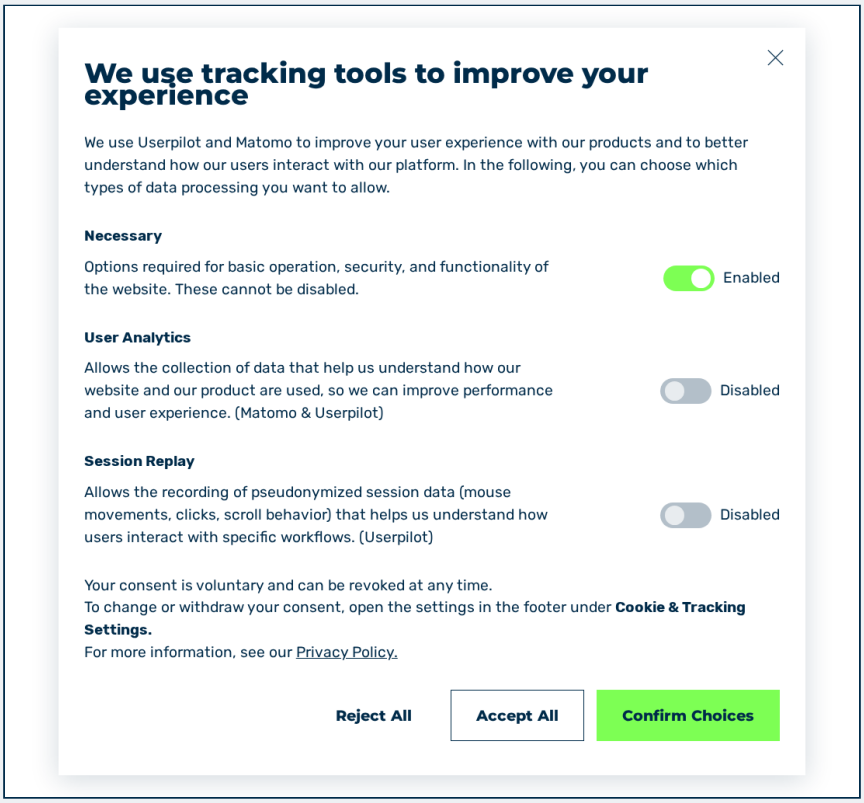


Abbildung 100: Dialog mit den Schaltern der drei Kategorien

- ▶ Ändern Sie die Schalter der Kategorien.
- ▶ Klicken Sie auf die Schaltfläche **Confirm Choices**.
- ➔ Das System übernimmt die geänderte Auswahl sofort.

Einzelheiten zur Verarbeitung nennt die **Datenschutzerklärung** der Myra Security GmbH.

4.4 Organisation

4.4.1 Mitglied einladen

Sie laden weitere Personen in Ihre Organisation ein. Das eingeladene Konto erhält eine E-Mail und bestätigt darüber seine Aufnahme.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Ihr Konto hat die Rolle **Org Admin**.

☑ Sie kennen die E-Mail-Adresse der einzuladenden Person.

Um ein Mitglied einzuladen, gehen Sie wie folgt vor:

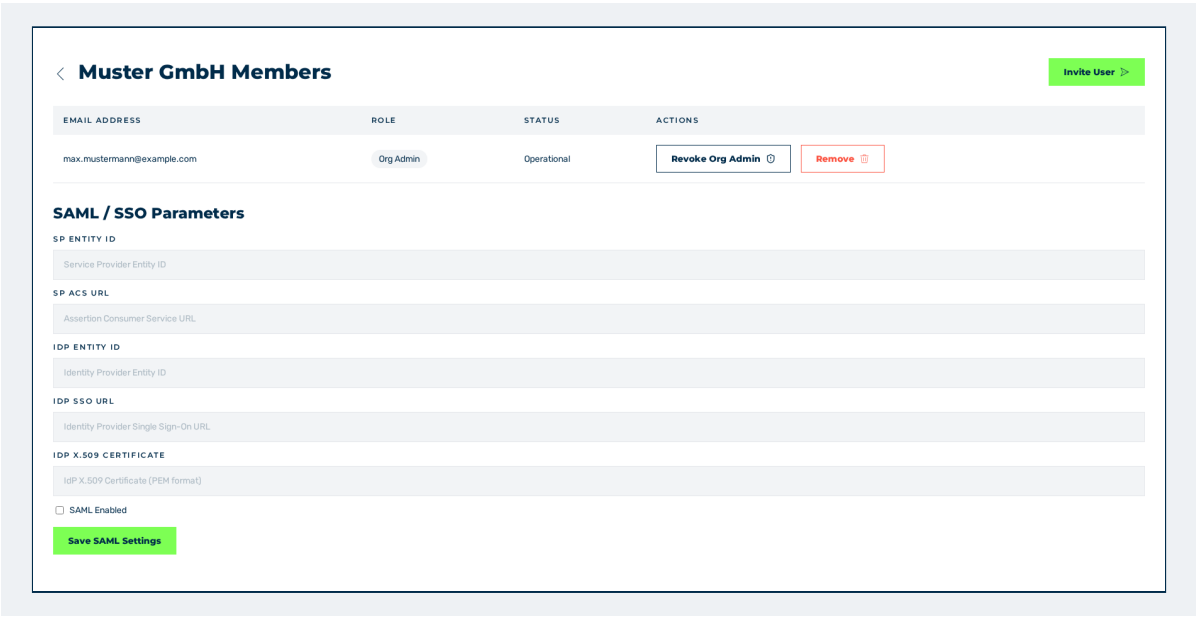


Abbildung 101: Ansicht Organization

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile** und klicken Sie im Bereich **Organization** auf die Schaltfläche **Manage Members**.

► Klicken Sie auf die Schaltfläche **Invite User**.

↳ Der Dialog **Invite User** wird angezeigt.

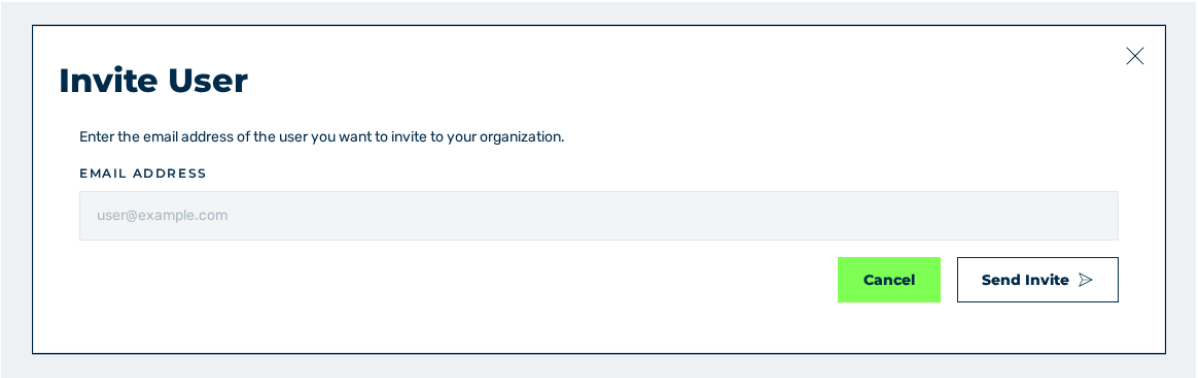


Abbildung 102: Dialog Invite User

- ▶ Geben Sie in das Feld **Email address** die E-Mail-Adresse ein.
- ▶ Klicken Sie auf die Schaltfläche **Send Invite**.
- ➔ Das System versendet die Einladung und zeigt die Meldung **Invitation sent.** an. Der Eintrag erscheint mit dem Status **Pending Verification** in der Tabelle.

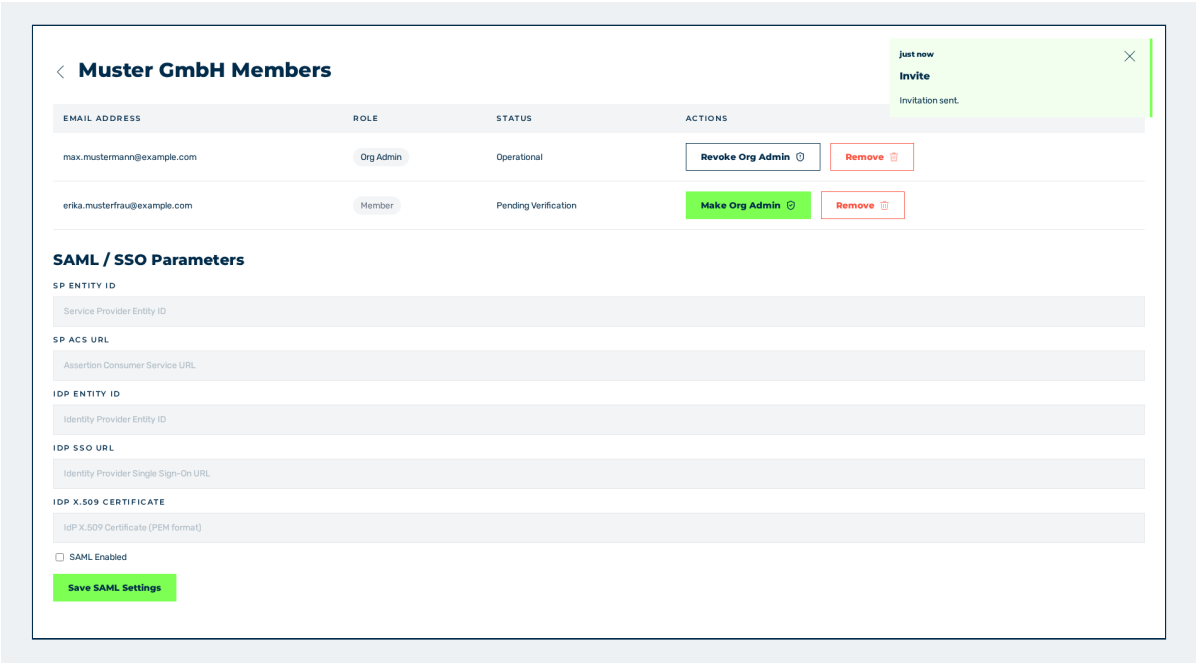


Abbildung 103: Ansicht Organization mit dem neuen Eintrag

Sobald die Person die Einladung bestätigt hat, wechselt der Status auf **Operational**.

Siehe **Ansicht Organization**.

4.4.2 Rolle Org Admin vergeben und entziehen

Die Rolle **Org Admin** berechtigt ein Mitglied, weitere Mitglieder zu verwalten und die Einstellungen für SAML und SSO zu ändern.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Ihr Konto hat die Rolle **Org Admin**.

☑ Das Mitglied gehört bereits zur Organisation.

4.4.2.1 Rolle vergeben

Um die Rolle zu vergeben, gehen Sie wie folgt vor:

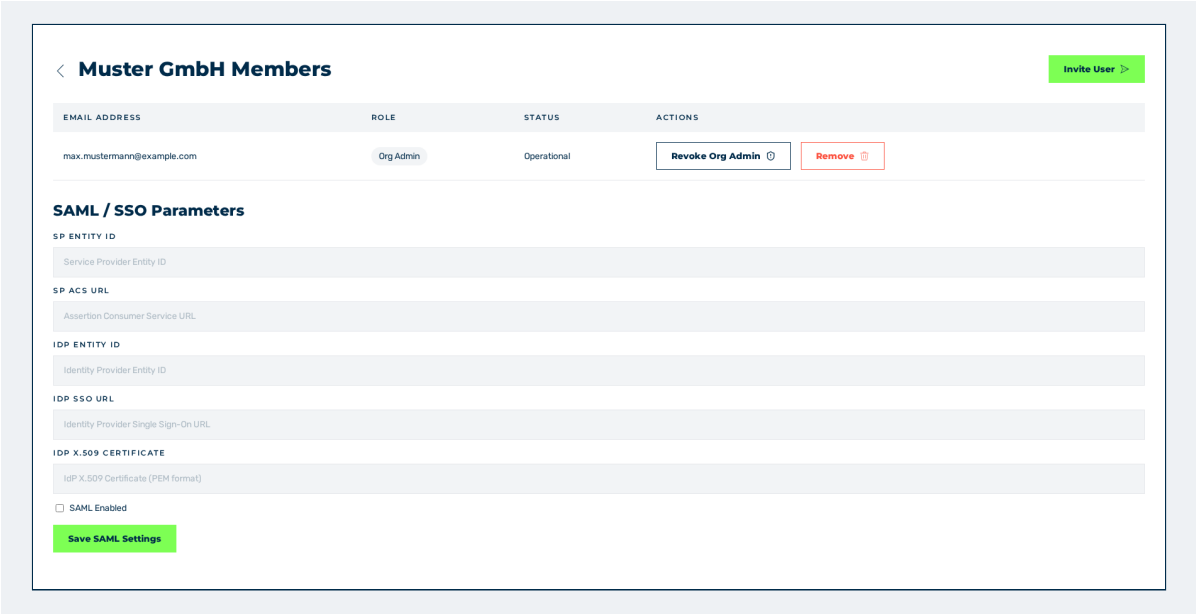


Abbildung 104: Ansicht Organization

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile** und klicken Sie im Bereich **Organization** auf die Schaltfläche **Manage Members**.

► Klicken Sie in der Zeile des Mitglieds in der Spalte **Actions** auf die Schaltfläche **Make Org Admin**.

↳ Der Dialog **Grant Org Admin** wird angezeigt.

Abbildung 105: Dialog Grant Org Admin

- ► Klicken Sie auf die Schaltfläche **Confirm**.
- ➔ Das System vergibt die Rolle und schließt den Dialog. Die Spalte **Role** zeigt den Wert **Org Admin** an.

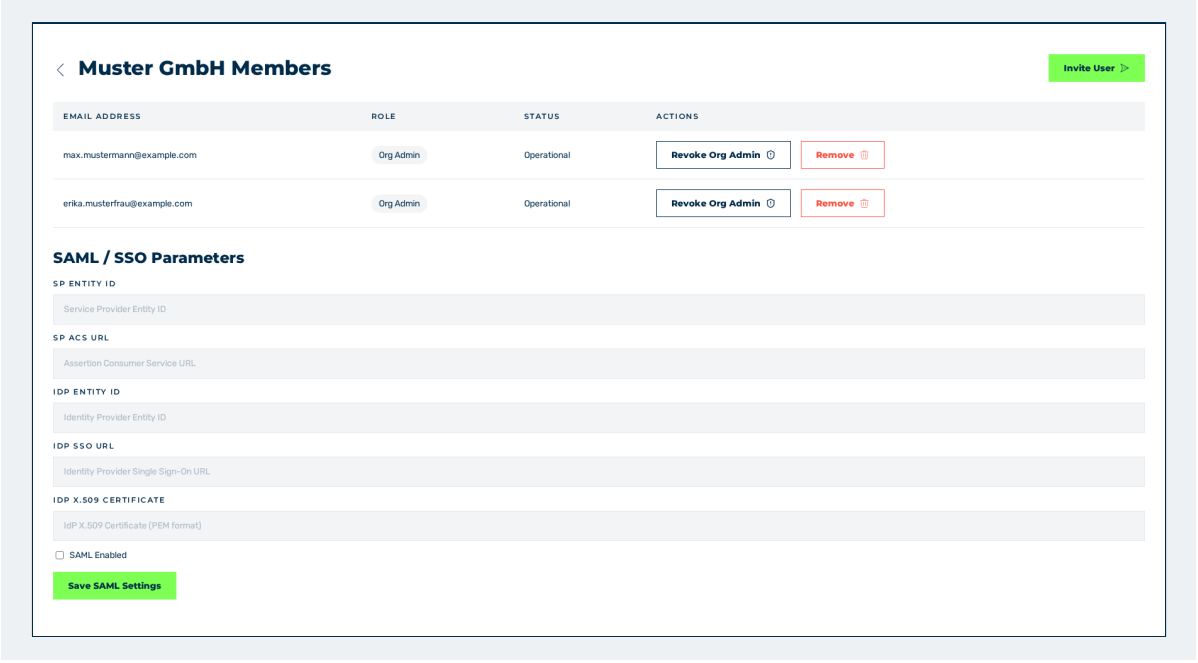


Abbildung 106: Ansicht Organization mit der vergebenen Rolle

Um den Vorgang abzubrechen, klicken Sie auf die Schaltfläche **Close**.

4.4.2.2 Rolle entziehen

Um die Rolle zu entziehen, gehen Sie wie folgt vor:

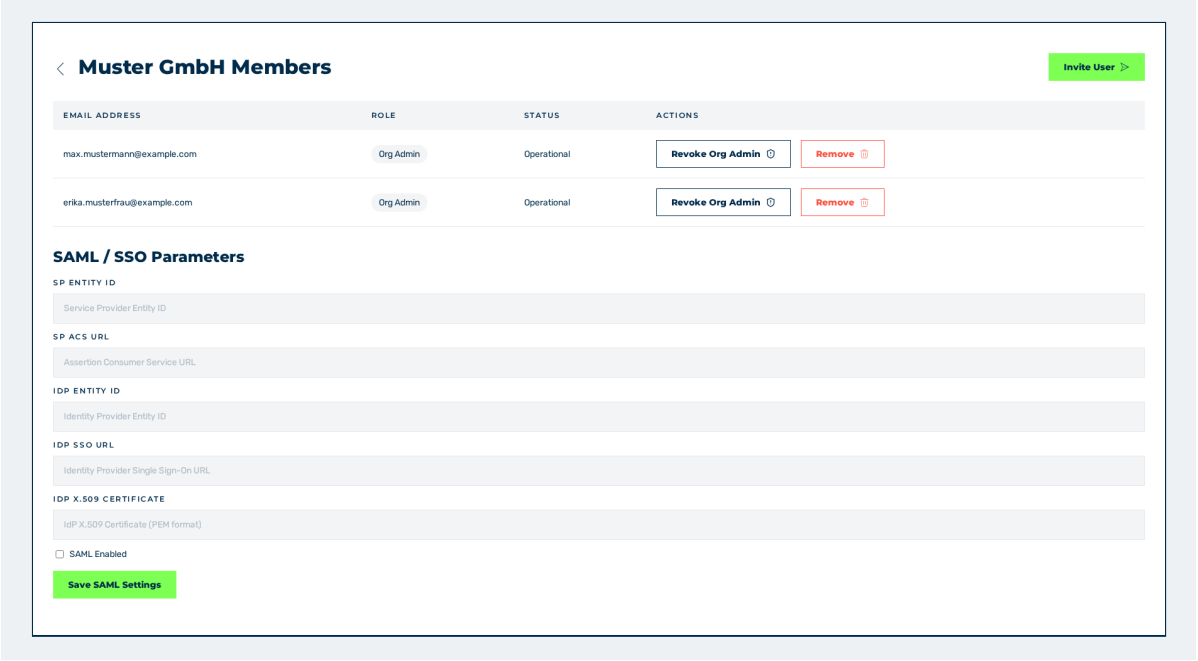


Abbildung 107: Ansicht Organization mit der vergebenen Rolle

► Klicken Sie in der Zeile des Mitglieds in der Spalte **Actions** auf die Schaltfläche **Revoke Org Admin**.

↳ Der Dialog **Revoke Org Admin** wird angezeigt.

Abbildung 108: Dialog Revoke Org Admin

► Klicken Sie auf die Schaltfläche **Confirm**.

➔ Das System entzieht die Rolle und schließt den Dialog. Die Spalte **Role** zeigt den Wert **Member** an.

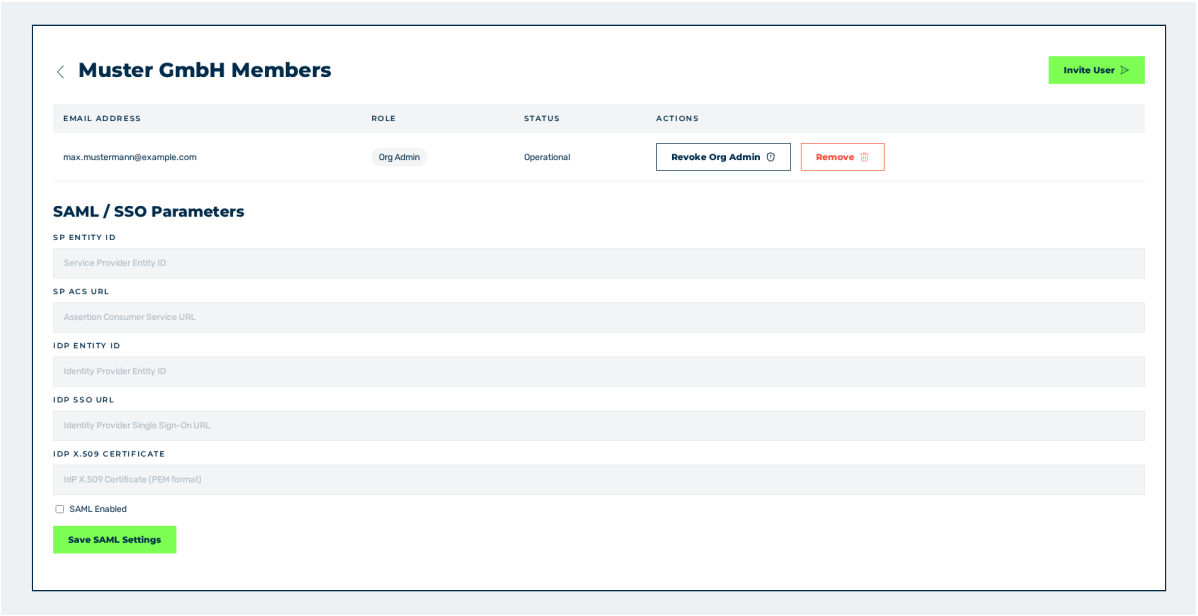


Abbildung 109: Ansicht Organization ohne die Rolle

Siehe **Mitglied entfernen**.

Siehe **Ansicht Organization**.

4.4.3 Mitglied entfernen

Sie entfernen ein Mitglied aus Ihrer Organisation. Das Konto verliert damit den Zugriff auf die Sitekeys der Organisation.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Ihr Konto hat die Rolle **Org Admin**.

☑ Das Mitglied gehört zur Organisation.

Um ein Mitglied zu entfernen, gehen Sie wie folgt vor:

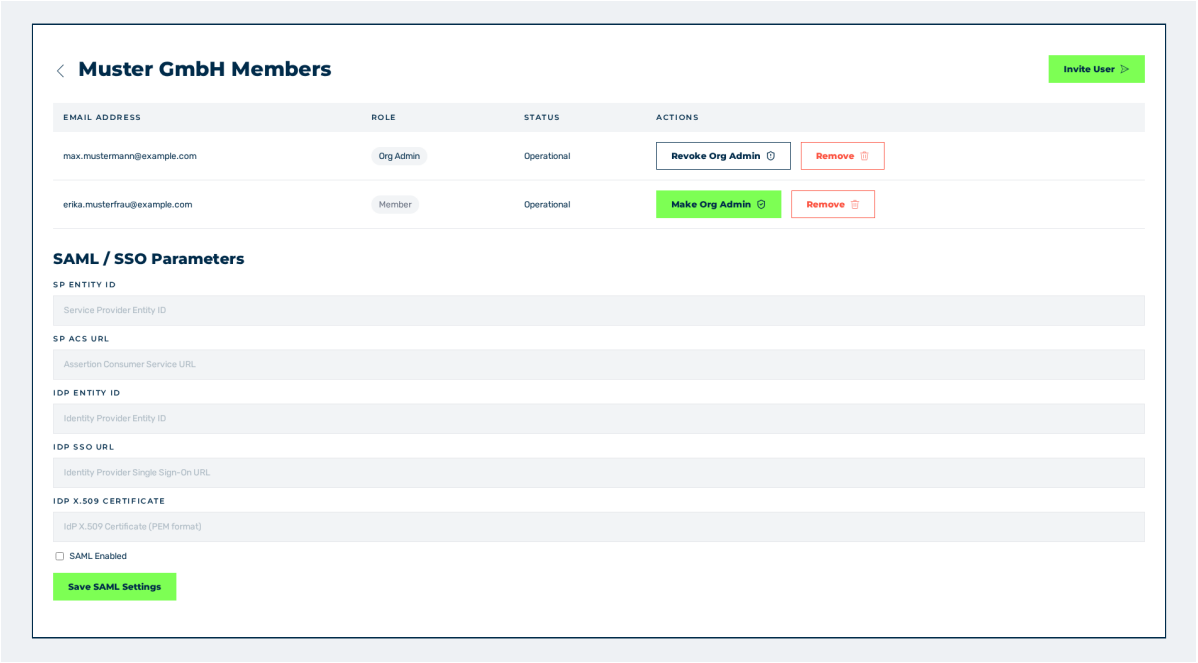


Abbildung 110: Ansicht Organization mit zwei Mitgliedern

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile** und klicken Sie im Bereich **Organization** auf die Schaltfläche **Manage Members**.

► Klicken Sie in der Zeile des Mitglieds in der Spalte **Actions** auf die Schaltfläche **Remove**.

↳ Der Dialog **Remove Member** wird angezeigt. Der Dialog nennt die E-Mail-Adresse des Mitglieds.

Abbildung 111: Dialog Remove Member

- ► Klicken Sie auf die Schaltfläche **Confirm**.
- ➔ Das System entfernt das Mitglied und schließt den Dialog. Der Eintrag verschwindet aus der Tabelle.

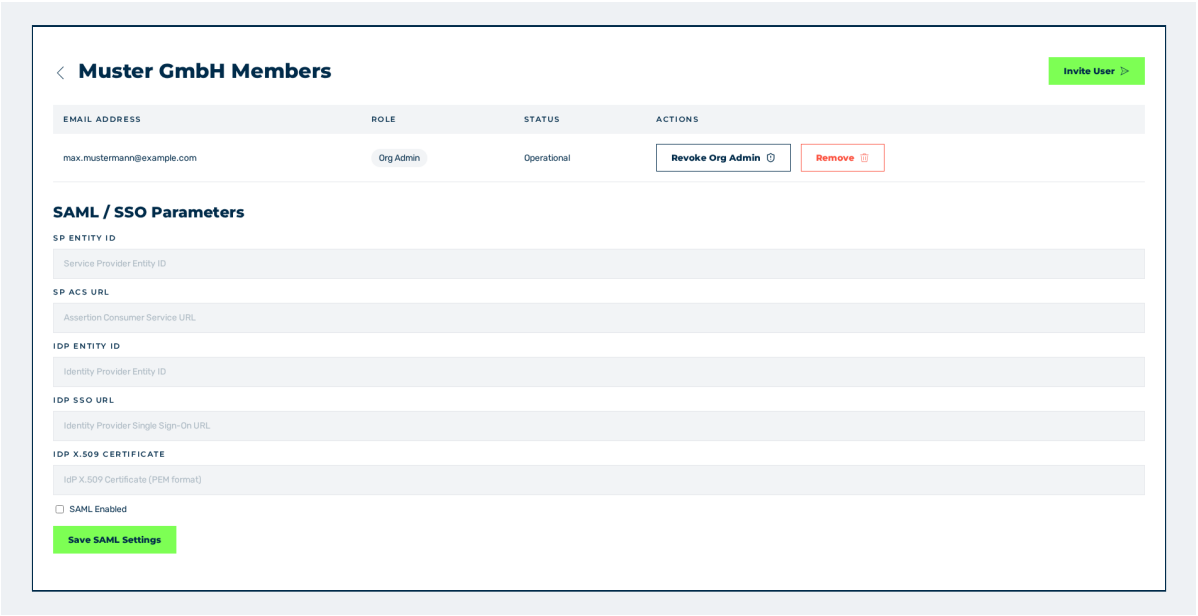


Abbildung 112: Ansicht Organization ohne das entfernte Mitglied

Um den Vorgang abzubrechen, klicken Sie auf die Schaltfläche **Close**.

Siehe **Rolle Org Admin vergeben und entziehen**.

Siehe **Ansicht Organization**.

4.4.4 SAML einrichten

Über SAML melden sich die Mitglieder Ihrer Organisation mit den Zugangsdaten Ihres eigenen Identitätsanbieters am Kundenportal an.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Ihr Konto hat die Rolle **Org Admin**.

☑ Sie verwalten einen Identitätsanbieter mit Unterstützung für SAML 2.0.

☑ Sie kennen die Kennung, die Anmeldeadresse und das Zertifikat Ihres Identitätsanbieters.

Um SAML einzurichten, gehen Sie wie folgt vor:

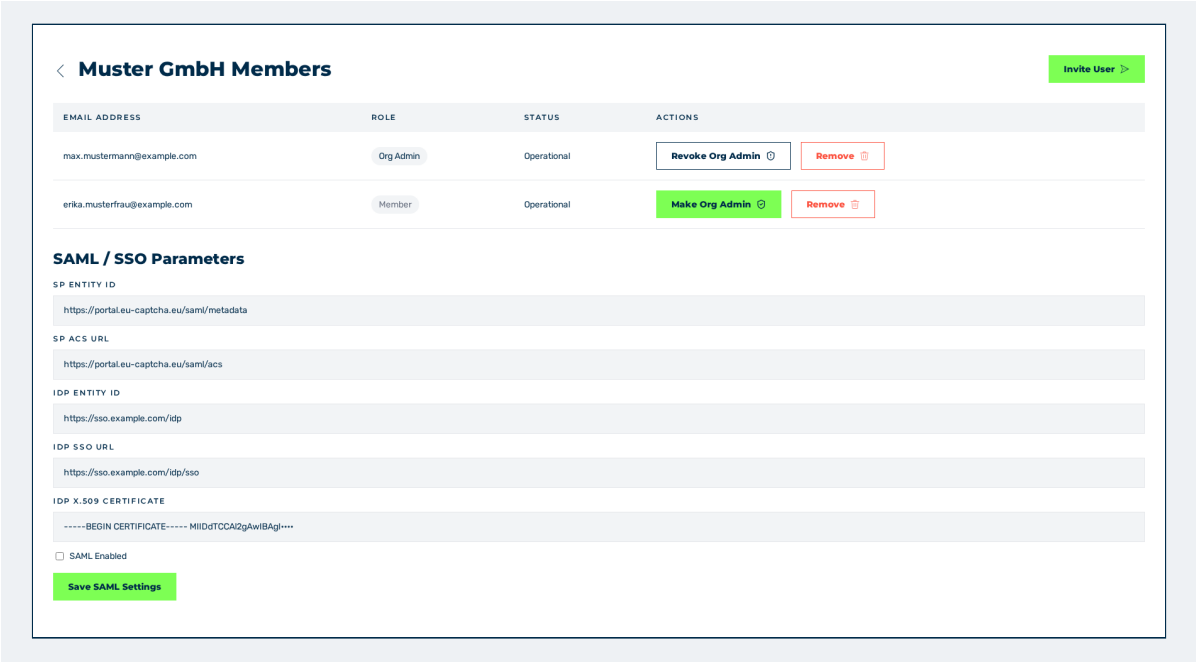


Abbildung 113: Bereich SAML / SSO Parameters

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile** und klicken Sie im Bereich **Organization** auf die Schaltfläche **Manage Members**.

► Blättern Sie zum Bereich **SAML / SSO Parameters**.

► Übernehmen Sie die Werte der Felder **SP Entity ID** und **SP ACS URL** in die Konfiguration Ihres Identitätsanbieters.

► Geben Sie in das Feld **IdP Entity ID** die Kennung Ihres Identitätsanbieters ein.

► Geben Sie in das Feld **IdP SSO URL** die Anmeldeadresse Ihres Identitätsanbieters ein.

- ▶ Fügen Sie in das Feld **IdP X.509 Certificate** das Zertifikat im Format PEM ein.
- ▶ Aktivieren Sie das Kontrollkästchen **SAML Enabled**.
- ▶ Klicken Sie auf die Schaltfläche **Save SAML Settings**.
- ➔ Das System speichert die Angaben und zeigt die Meldung **SAML Settings** mit dem Text **SAML settings saved successfully.** an.

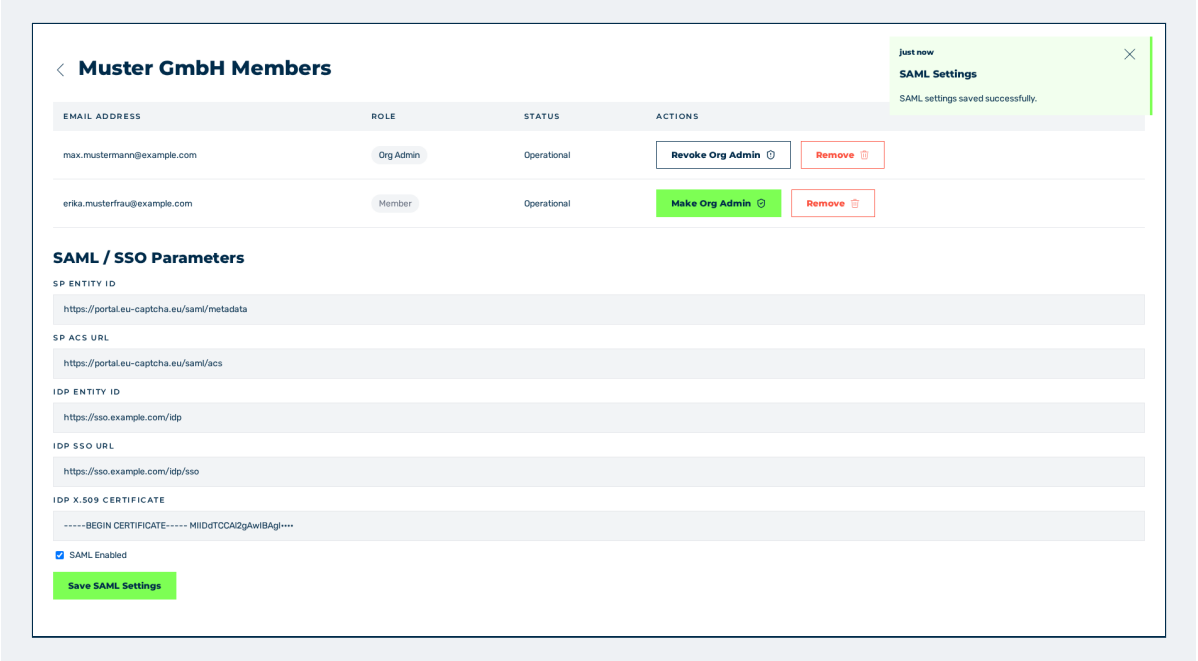


Abbildung 114: Bereich SAML / SSO Parameters nach dem Speichern

- ▶ Prüfen Sie die Anmeldung über Ihren Identitätsanbieter mit einem Testkonto.

4.5 Abonnement

4.5.1 Plan buchen

Mit einem kostenpflichtigen Plan stehen die Einstellungen der Sitekeys und der persönliche Support zur Verfügung. Die Abrechnung läuft über einen Zahlungsdienstleister.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet.

☑ Für Ihr Konto ist kein Abonnement aktiv.

☑ Ihnen liegen die Rechnungsdaten und ein Zahlungsmittel vor.

Um einen Plan zu buchen, gehen Sie wie folgt vor:

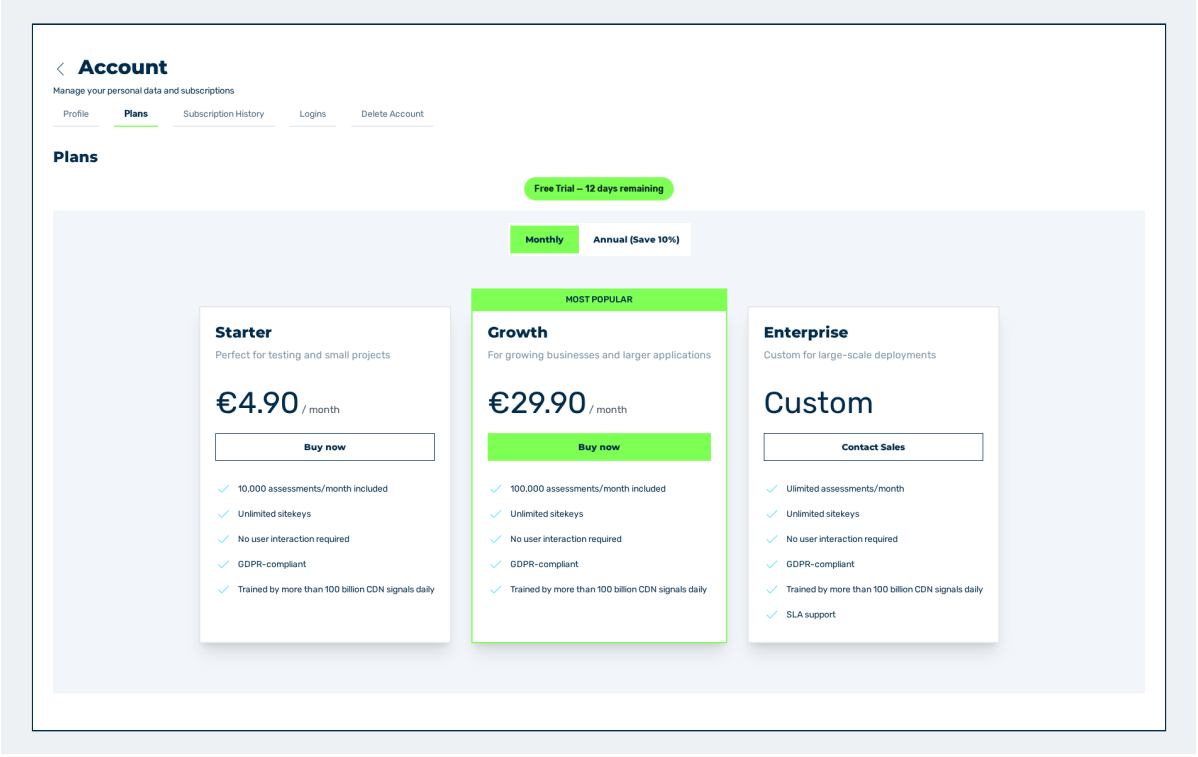


Abbildung 115: Reiter Plans

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile** und wechseln Sie in den Reiter **Plans**.

► Klicken Sie auf die Schaltfläche **Monthly** oder **Annual (Save 10%)**.

↳ Die Preise der Pläne wechseln auf den gewählten Abrechnungszeitraum.

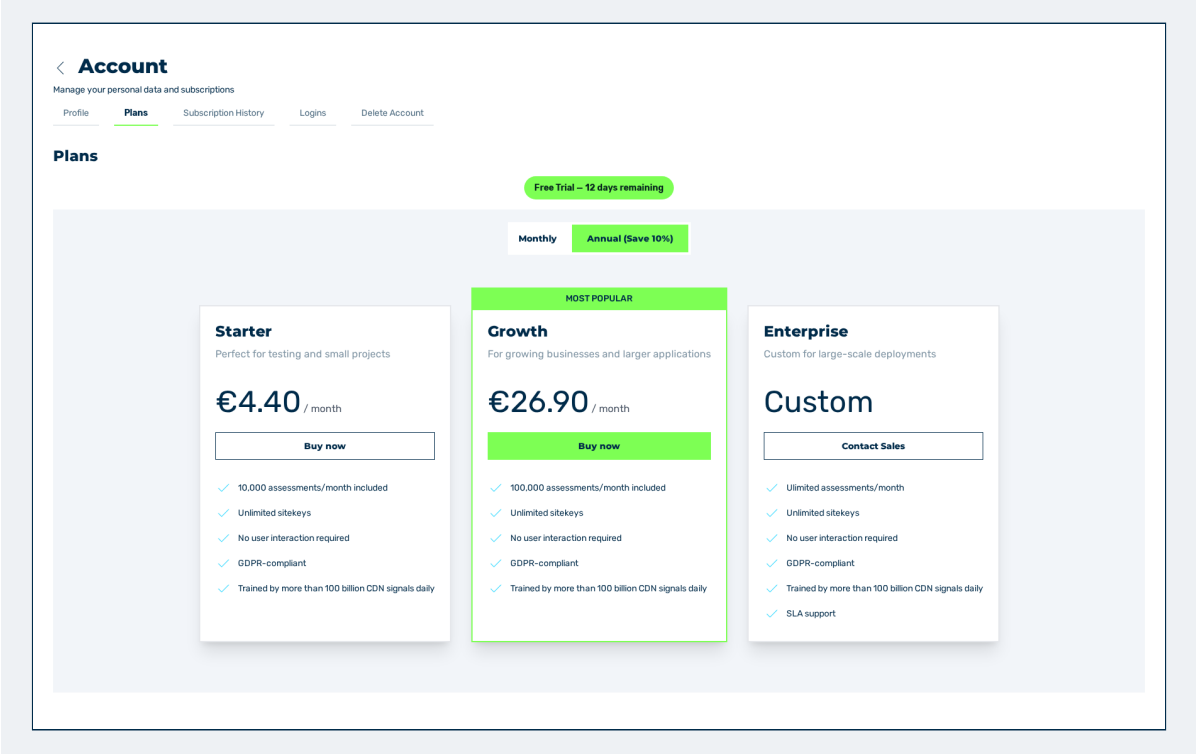


Abbildung 116: Reiter Plans mit jährlicher Abrechnung

► Klicken Sie im gewünschten Plan auf die Schaltfläche **Buy now**.

↳ Die Ansicht **Complete Your Purchase** wird angezeigt.

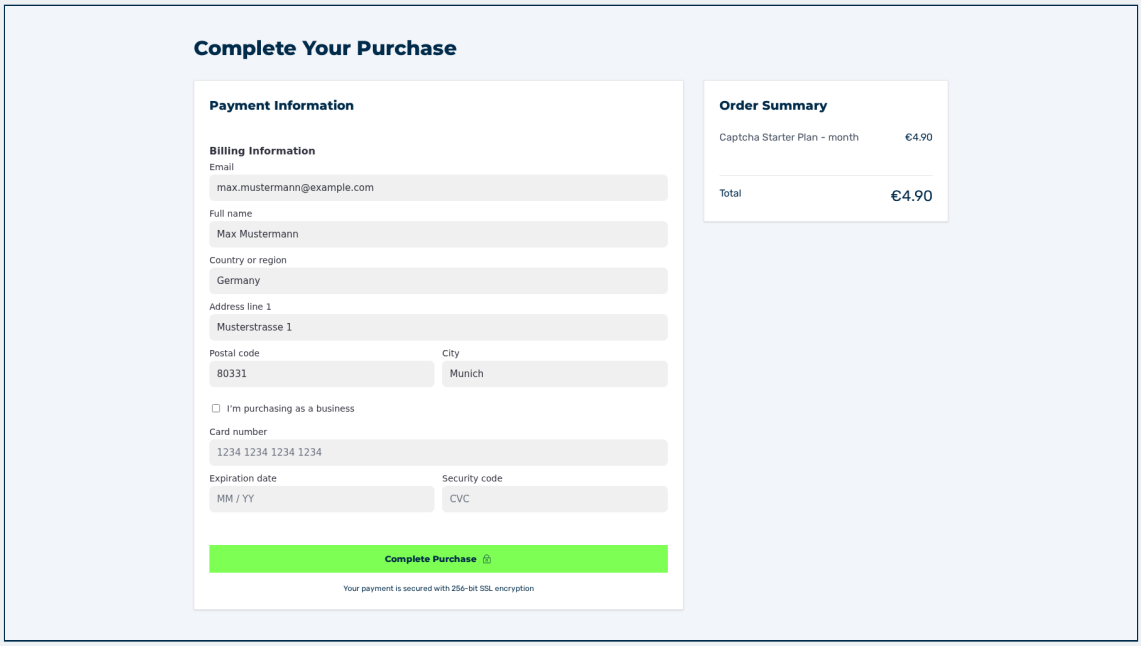


Abbildung 117: Ansicht Complete Your Purchase

- ▶ Geben Sie im Bereich **Billing Information** die Rechnungsdaten ein.
- ▶ Geben Sie im Bereich **Payment Information** die Daten Ihres Zahlungsmittels ein.
- ▶ Klicken Sie auf die Schaltfläche **Complete Purchase**.

➔ Das System aktiviert den Plan. Der Reiter **Subscription History** enthält einen Eintrag mit dem Status des Abonnements.

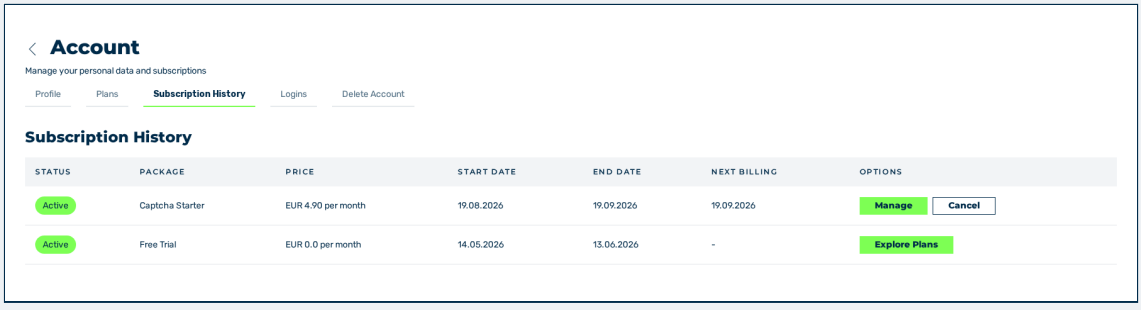


Abbildung 118: Reiter Subscription History mit dem neuen Abonnement

4.5.1.1 Buchung als Unternehmen

Als Unternehmen aktivieren Sie das Kontrollkästchen **I'm purchasing as a business**. Die Ansicht zeigt daraufhin einen Hinweis und das Feld **VAT Number (optional)** an.

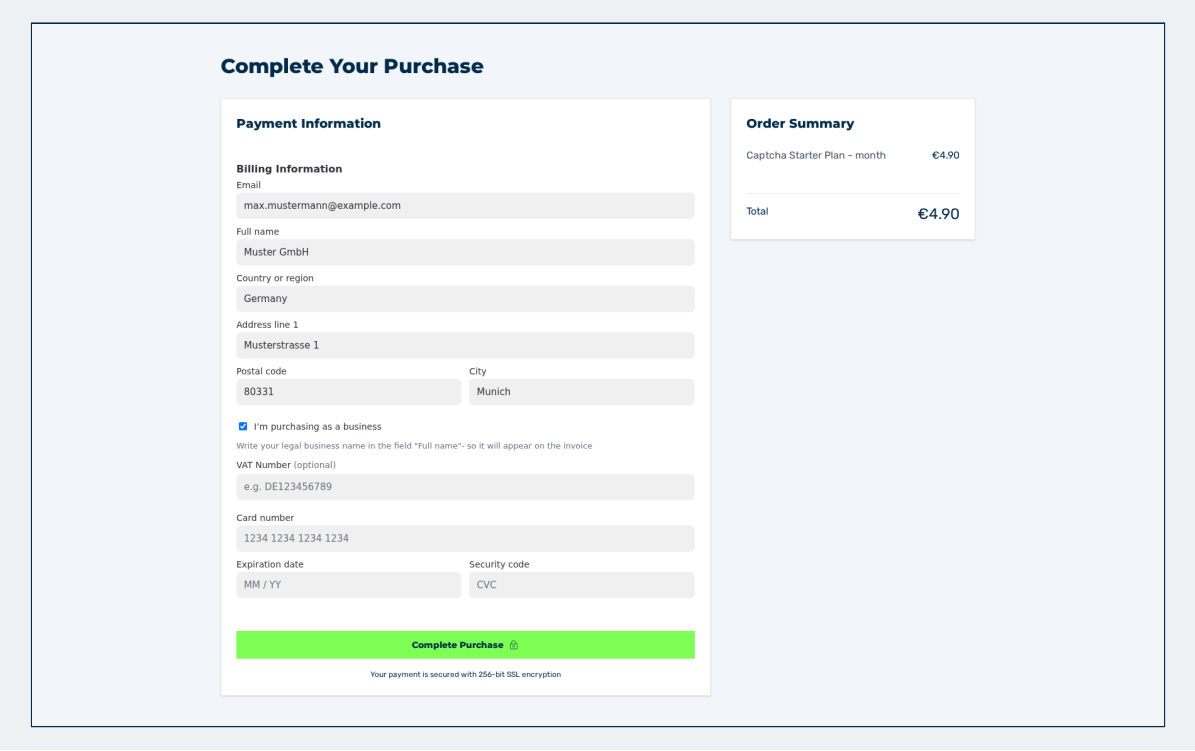


Abbildung 119: Ansicht Complete Your Purchase mit den Angaben eines Unternehmens

- ▶ Aktivieren Sie das Kontrollkästchen **I'm purchasing as a business**.
  - ↳ Ein Hinweis zum Firmennamen im Feld **Full name** und das Feld **VAT Number (optional)** werden angezeigt.
- ▶ Geben Sie in das Feld **Full name** den Namen Ihres Unternehmens ein.
- ▶ Geben Sie in das Feld **VAT Number (optional)** Ihre Umsatzsteuer-Identifikationsnummer ein, zum Beispiel `DE123456789` .
- ➔ Die Rechnung trägt den Namen des Unternehmens. Mit einer Umsatzsteuer-Identifikationsnummer aus der EU rechnet das System im Reverse-Charge-Verfahren ab.

4.5.1.2 Plan Enterprise

Für den Plan **Enterprise** führt die Schaltfläche **Contact Sales** auf das Kontaktformular von Myra. Der Preis und der Umfang werden im Einzelfall vereinbart.

4.5.1.3 Abonnement verwalten

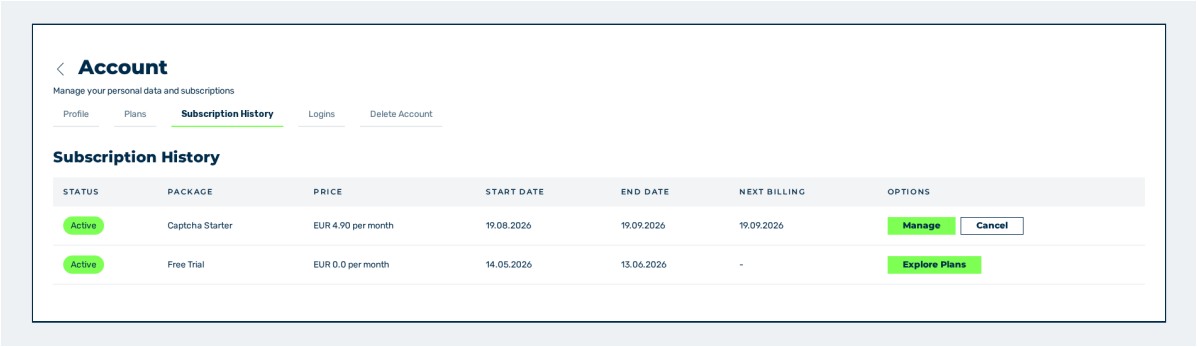


Abbildung 120: Reiter Subscription History mit der Schaltfläche Manage

Zahlungsdaten und Rechnungen verwalten Sie über die Schaltfläche **Manage** im Reiter **Subscription History**. Die Schaltfläche öffnet das Kundenportal des Zahlungsdienstleisters.

Siehe **Reiter Subscription History**.

Siehe **Abonnement kündigen**.

4.5.2 Abonnement kündigen

Sie kündigen ein Abonnement im Reiter **Subscription History**. Das Abonnement bleibt bis zum Ende des bezahlten Zeitraums aktiv. Danach erfolgt keine weitere Abrechnung.

Die folgenden Voraussetzungen müssen erfüllt sein:

☑ Sie sind am Kundenportal angemeldet.

☑ Für Ihr Konto ist ein Abonnement aktiv.

Um ein Abonnement zu kündigen, gehen Sie wie folgt vor:

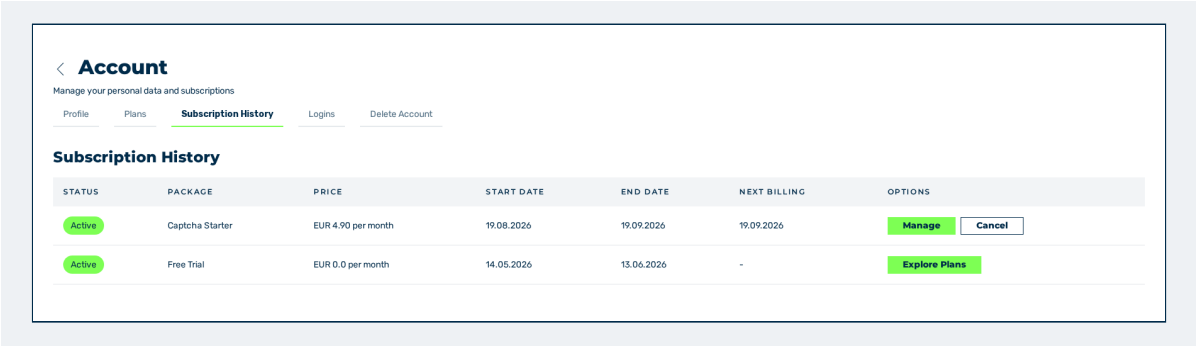


Abbildung 121: Reiter Subscription History mit einem gebuchten Plan

► Öffnen Sie über die Seitenleiste den Eintrag **Account** › **Profile** und wechseln Sie in den Reiter **Subscription History**.

► Klicken Sie in der Zeile des Abonnements auf die Schaltfläche **Cancel**.

↳ Der Dialog **Cancel subscription** wird angezeigt. Der Dialog nennt den Plan und das Ende des bezahlten Zeitraums.

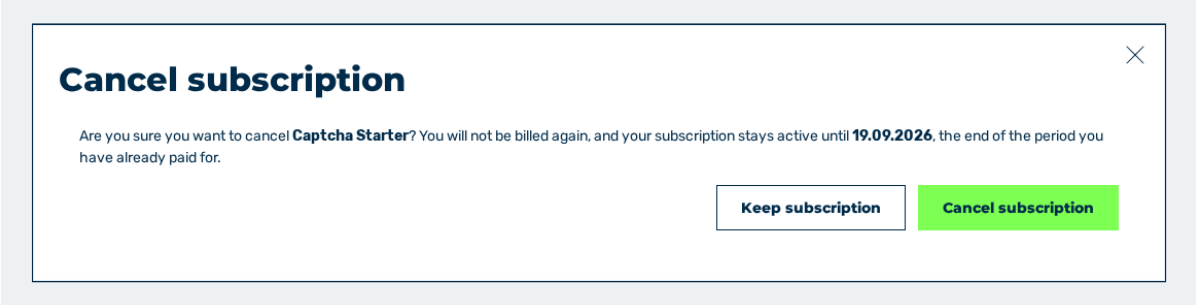


Abbildung 122: Dialog Cancel subscription

► Klicken Sie auf die Schaltfläche **Cancel subscription**.

➔ Das System kündigt das Abonnement. Die Spalte **Status** zeigt die Marke **Cancelled** an, die Spalte **Next billing** den Wert `Ends` mit dem Ende des bezahlten Zeitraums.

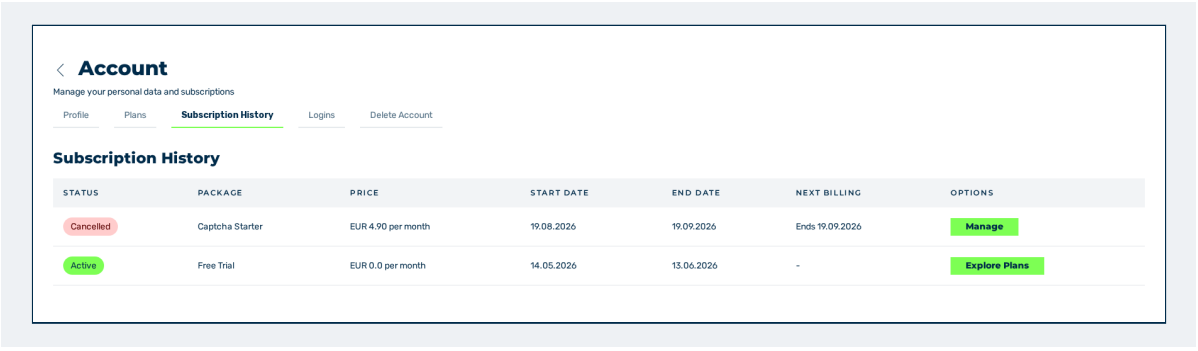


Abbildung 123: Reiter Subscription History mit dem gekündigten Abonnement

4.5.2.1 Buchung während der Testphase kündigen

Ein Plan, den Sie während der Testphase gebucht haben, trägt die Marke **Committed** und startet erst nach dem Ende der Testphase. Diese Buchung kündigen Sie über die Schaltfläche **Cancel commitment**.

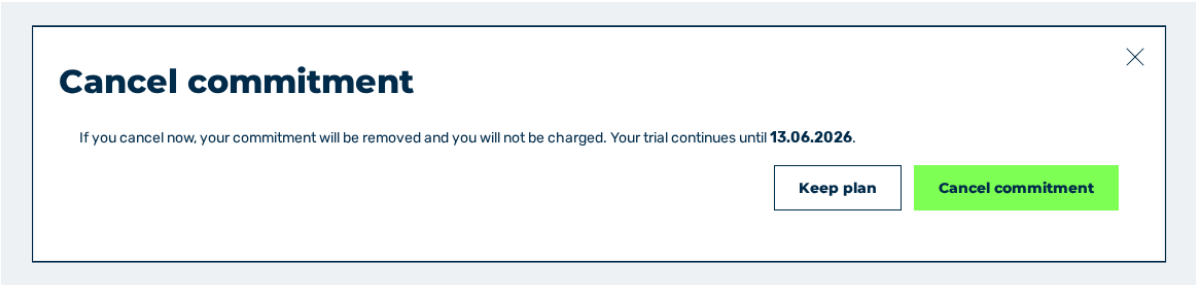


Abbildung 124: Dialog Cancel commitment

- ▶ Klicken Sie in der Zeile des Abonnements auf die Schaltfläche **Cancel commitment**.
  - ↳ Der Dialog **Cancel commitment** wird angezeigt.
- ▶ Klicken Sie auf die Schaltfläche **Cancel commitment**.
- ➔ Das System entfernt die Buchung. Es erfolgt keine Abrechnung. Die Testphase läuft bis zu ihrem Ende weiter.

> Die Schaltflächen **Keep subscription** und **Keep plan** schließen den Dialog, ohne zu kündigen.

Siehe Reiter Subscription History.

Siehe Konto löschen.

5. Integration

5.1 Allgemein

5.1.1 Integration

Die Integration des Myra EU CAPTCHA besteht immer aus zwei Teilen: dem Widget in Ihrer Seite und der Prüfung des Tokens auf Ihrem Server. Erst beide Teile zusammen ergeben einen Schutz.

Für die verbreiteten Plattformen stehen Bibliotheken, Erweiterungen und Beispiele bereit, die diese beiden Teile übernehmen. Die folgenden Kapitel beschreiben je Plattform die Installation, die Konfiguration und die Prüfung.

Die Anleitungen dieses Kapitels setzen einen angelegten Sitekey voraus. Der Weg von einem neuen Konto bis zur geprüften Integration steht unter **Einrichtung** mit den Schritten **Ersten Sitekey anlegen**, **Widget einbinden** und **Integration testen**.

5.1.1.1 Aufbau einer Integration

Jede Integration führt die folgenden drei Schritte aus:

Schritt	Ort	Inhalt
1	Frontend	Das Skript `verify.js` vom CDN laden.
2	Frontend	Das Widget-Element mit dem öffentlichen Sitekey einfügen.
3	Backend	Das Token vor der Verarbeitung über den Endpunkt `/verify` prüfen.

Siehe **Widget einbinden**.

> Die beiden Schritte im Frontend laufen im Browser der Besuchenden ab. Wenn Ihre Website eine Content Security Policy sendet, dann muss diese die Adressen des Dienstes freigeben. Diese Voraussetzung gilt für jede Plattform dieses Kapitels. Siehe **Content Security Policy**.

5.1.1.2 Verfügbare Anleitungen

Die folgenden Anleitungen stehen zur Verfügung:

Kategorie	Plattformen
CMS	**WordPress**, **Joomla**, **Drupal**, **TYPO3**
Frontend	**HTML und JavaScript**, **React**, **Vue**, **Angular**, **Svelte**
Backend	**PHP**, **Symfony und Laravel**, **Python**, **Django**, **Ruby**, **Java**, **Client aus der Spezifikation**
Mobile Apps	**Android**, **iOS**, **Flutter**
Infrastruktur	**Caddy**, **CrowdSec**, **Traefik**

5.1.1.3 Vollständige Beispiele

Die Anleitungen je Plattform beschreiben jeweils eine Seite der Integration. Die Beispiele führen beide Seiten an einem durchgehenden Fall zusammen: Sie schützen das Kontaktformular einer bereits laufenden Anwendung und ändern dafür nur wenige Dateien.

Die folgenden Beispiele stehen zur Verfügung:

Beispiel	Frontend	Backend
React und PHP	React mit Vite	PHP
Vue und Python	Vue mit Vite	Python mit Flask
Angular und Java	Angular	Java mit Spring Boot
Svelte und Ruby	Svelte mit Vite	Ruby mit Sinatra

Beispiel	Frontend	Backend
HTML und Django	HTML ohne Framework	Django

5.1.1.4 Angaben aus dem Kundenportal

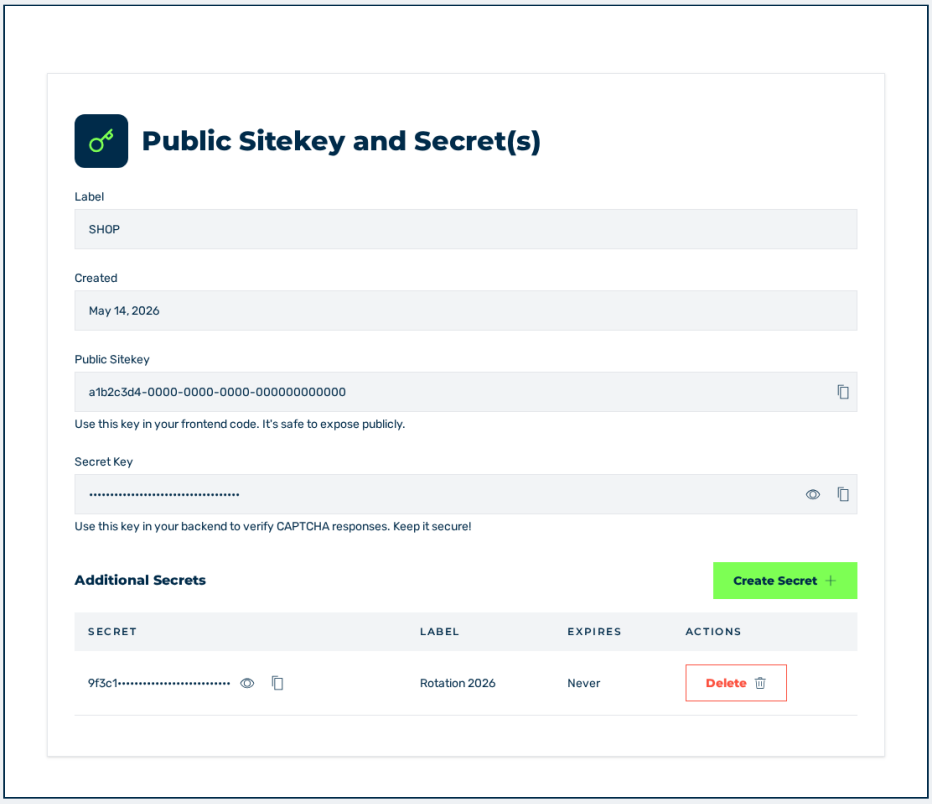


Abbildung 125: Ansicht Details

Alle Anleitungen benötigen die folgenden Angaben aus der Ansicht **Details** des Sitekeys:

Angabe	Verwendung
Public Sitekey	Im Widget-Element und in der Konfiguration des Frontends. Der Wert ist öffentlich.
Secret	Ausschließlich in der serverseitigen Prüfung. Der Wert bleibt auf dem Server.

Siehe Ansicht Details.

5.2 CMS

5.2.1 WordPress

Das Plugin **EU-Captcha** schützt die Formulare von WordPress und verbreiteter Formular-Plugins. Es übernimmt sowohl das Widget als auch die serverseitige Prüfung.

> **Video:** Einrichtung des Plugins in WordPress — abspielbar in der Online-Hilfe

5.2.1.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
WordPress	ab Version 5.8, geprüft bis Version 6.9
PHP	ab Version 7.4
Zugang	Administrationsbereich von WordPress

5.2.1.2 Geschützte Formulare

Das Plugin schützt die folgenden Formulare:

- Anmeldung von WordPress
- Registrierung von WordPress
- Kommentare von WordPress
- Contact Form 7
- Ninja Forms
- WooCommerce Checkout
- WPForms

5.2.1.3 Plugin einrichten

Um das Plugin einzurichten, gehen Sie wie folgt vor:

► Laden Sie den Ordner `eu-captcha` in das Verzeichnis `/wp-content/plugins/` Ihrer Installation.

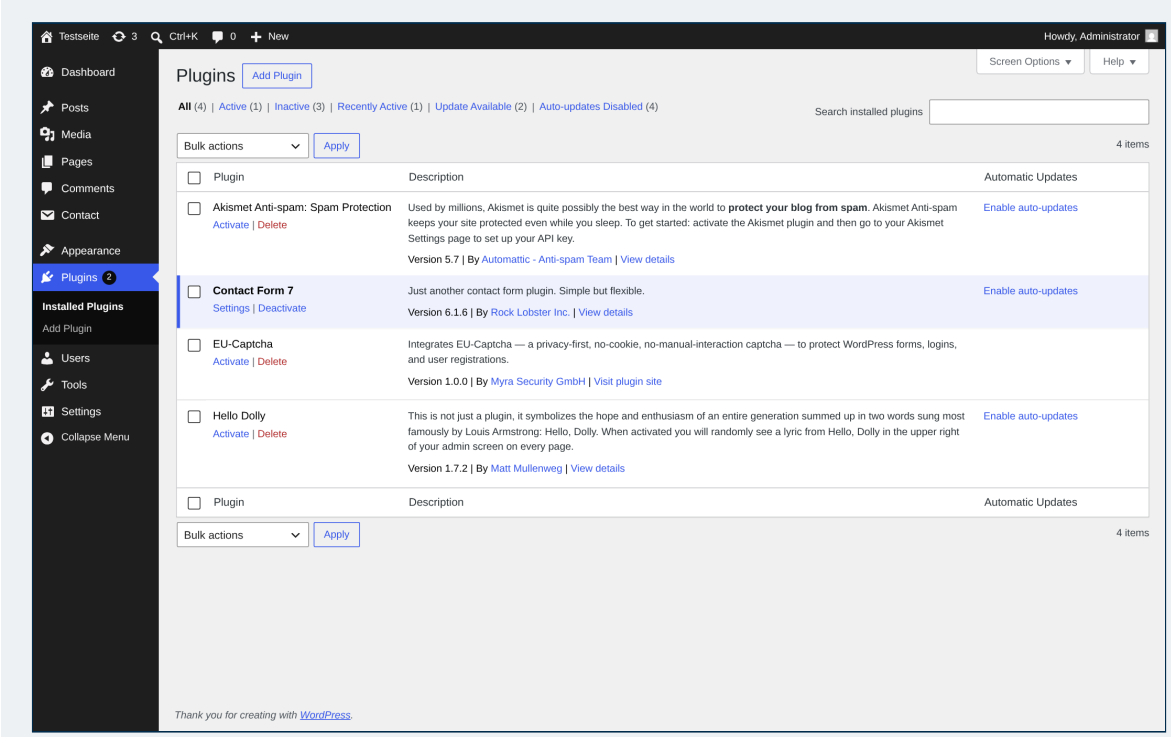


Abbildung 126: Liste der Plugins mit dem Plugin EU-Captcha

► Aktivieren Sie das Plugin über das Menü **Plugins**.

↳ Die Einstellungsseite **EU-Captcha** steht zur Verfügung.

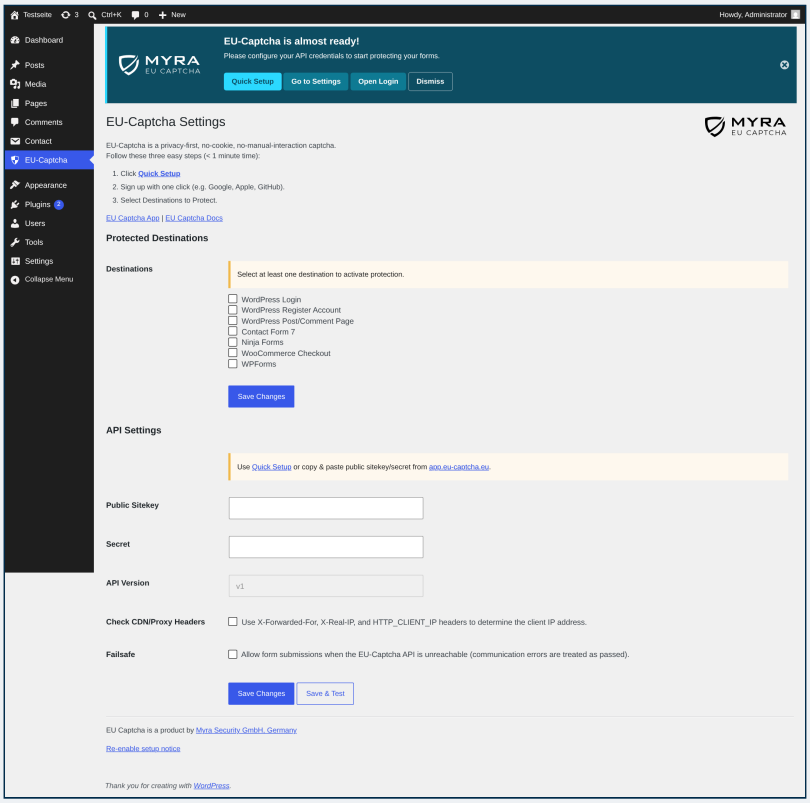


Abbildung 127: Einstellungsseite EU-Captcha

- ► Öffnen Sie die Einstellungsseite **EU-Captcha**.
- ► Klicken Sie auf die Schaltfläche **Quick Setup**. Alternativ tragen Sie den öffentlichen Sitekey und das Secret von Hand ein.
- ► Wählen Sie unter **Protected Destinations** die zu schützenden Formulare aus.
- ► Klicken Sie auf die Schaltfläche **Save Changes**.
- ➔ Das Plugin bindet das Widget in die gewählten Formulare ein und prüft die Token serverseitig.

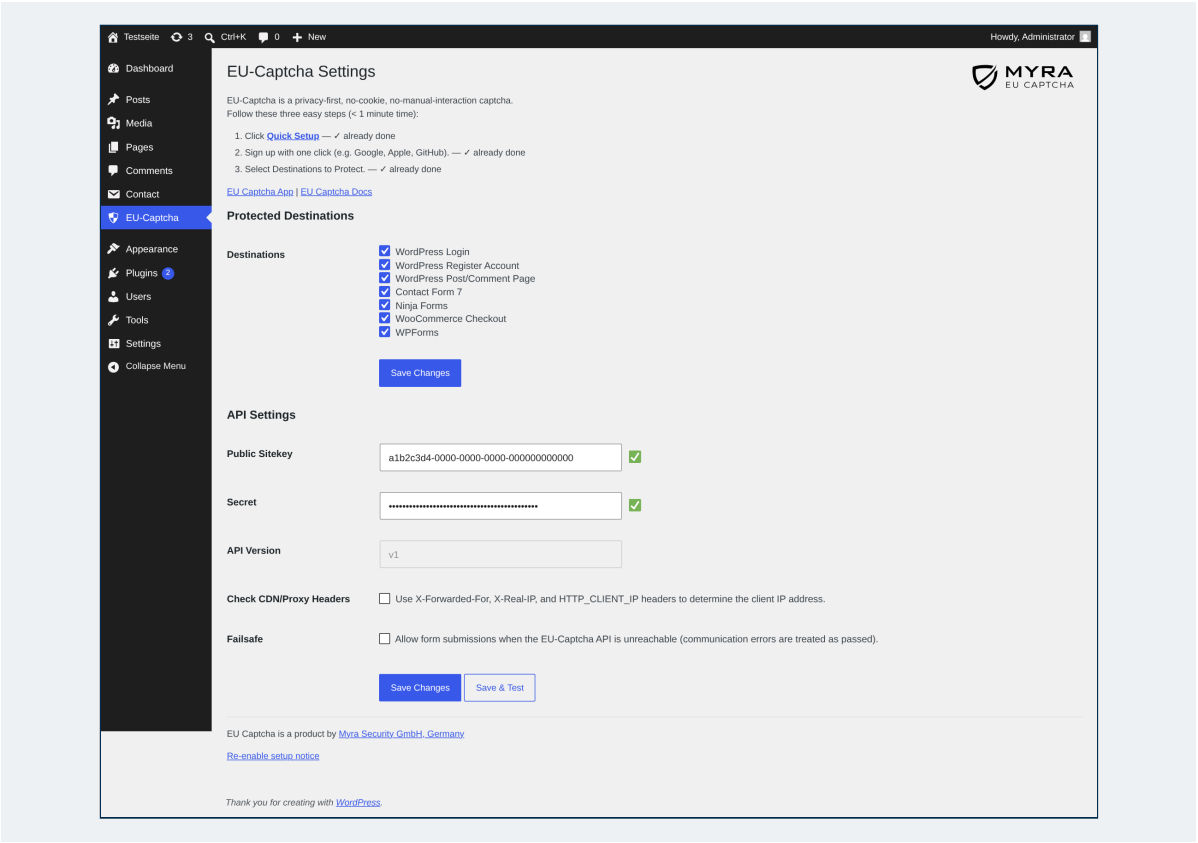


Abbildung 128: Einstellungsseite EU-Captcha nach dem Speichern

5.2.1.4 Verhalten bei Störungen

In der Voreinstellung weist das Plugin eine Absendung zurück, wenn die API nicht erreichbar ist. Damit lässt sich der Schutz während einer Störung nicht umgehen.

Die Option **Failsafe** in den Einstellungen unter **API Settings** kehrt dieses Verhalten um: Übertragungsfehler wie Zeitüberschreitungen oder Namensauflösungsfehler gelten dann als bestandene Prüfung. Ungültige Antworten weist das Plugin weiterhin zurück.

> Mit aktivierter Option **Failsafe** bleibt Ihr Formular während einer Störung erreichbar, ist in dieser Zeit jedoch ungeschützt.

5.2.1.5 Eigene Formulare schützen

Das Plugin stellt die Funktion `eu_captcha_verify()` bereit. Andere Plugins und Themes prüfen damit ein Token mit den bereits hinterlegten Zugangsdaten.

Fügen Sie das Widget-Element in Ihr Formular ein:

```
<div class="eu-captcha" data-sitekey="a1b2c3d4-0000-0000-0000-000000000000"></
div>
```

Prüfen Sie das Token beim Absenden:

```
if ( function_exists( 'eu_captcha_verify' ) ) {
    $token  = sanitize_text_field( $_POST['eu-captcha-response'] );
    $result = eu_captcha_verify( $token );
    if ( ! $result ) {
        wp_die( 'Captcha verification failed.' );
    }
}
```

Die Funktion gibt `true` bei einer gültigen Prüfung zurück und `false` bei einem ungültigen Token, fehlenden Zugangsdaten oder einem Fehler der API. Prüfen Sie stets zuerst mit `function_exists()`, damit Ihr Quelltext auch bei deaktiviertem Plugin funktioniert.

Das Skript `verify.js` bindet das Plugin auf allen Seiten des Frontends und auf der Anmeldeseite selbst ein.

5.2.2 Joomla

Das Plugin für Joomla bindet sich in die Standard-CAPTCHA-Schnittstelle von Joomla ein. Jedes Formular mit einem CAPTCHA-Feld ist damit ohne eigenen Quelltext geschützt.

5.2.2.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Joomla	Version 4.x, 5.x oder 6.x
PHP	ab Version 8.1
Zugang	Administrationsbereich von Joomla

5.2.2.2 Plugin installieren

Um das Plugin zu installieren, gehen Sie wie folgt vor:

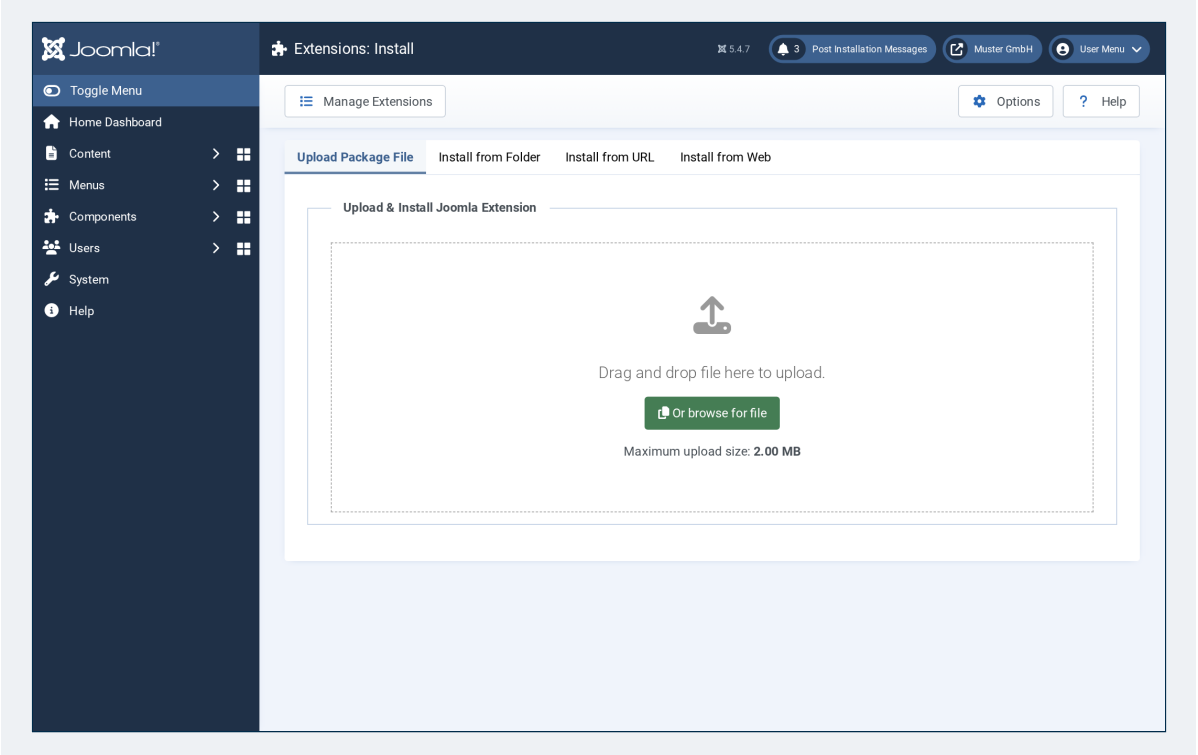


Abbildung 129: Ansicht Extensions: Install

- Laden Sie das Paket `pkg_eucaptcha.zip` herunter.
- Öffnen Sie im Administrationsbereich den Eintrag **Extensions** → **Install**.
- Laden Sie im Reiter **Upload Package File** die Datei `pkg_eucaptcha.zip` hoch.
  - ↳ Joomla installiert die beiden enthaltenen Plugins und meldet **Installation of the package was successful**.

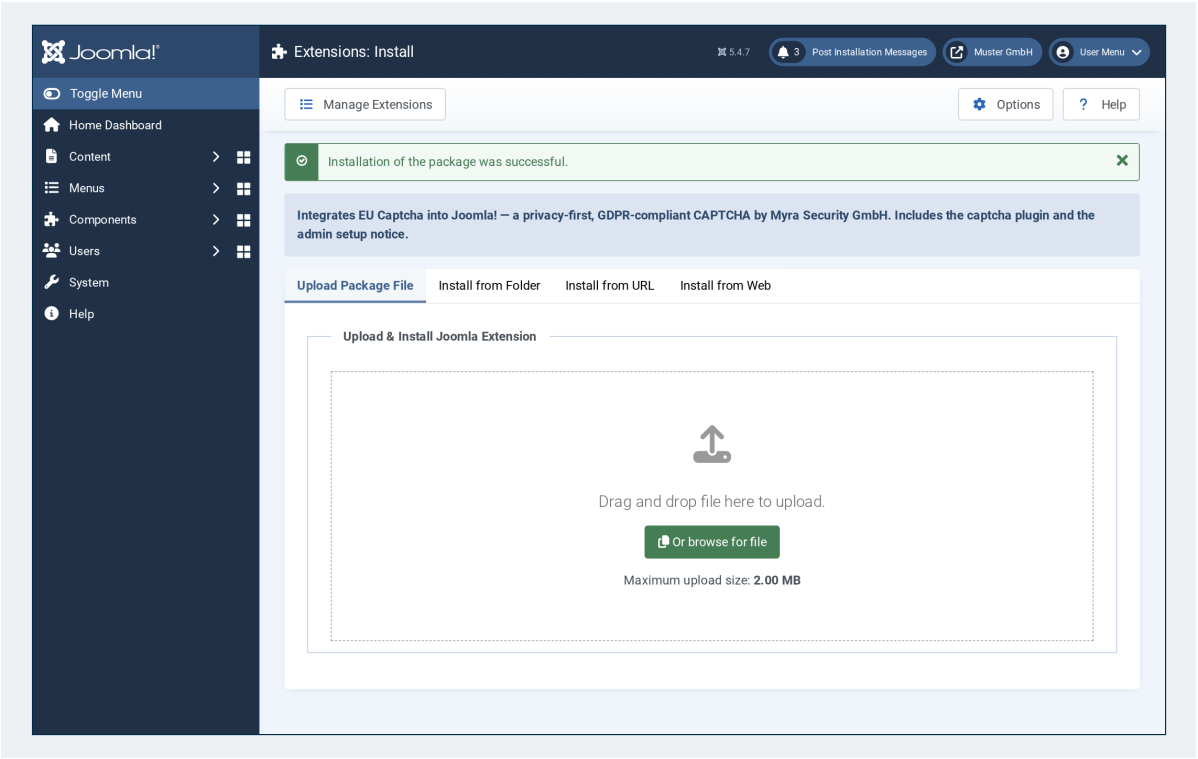


Abbildung 130: Meldung nach dem Installieren

► Öffnen Sie den Eintrag **Extensions** → **Plugins** und suchen Sie nach `eu captcha`.

↳ Die Liste zeigt die Plugins **EU Captcha** und **EU Captcha - Admin Notice**. Beide sind abgeschaltet.

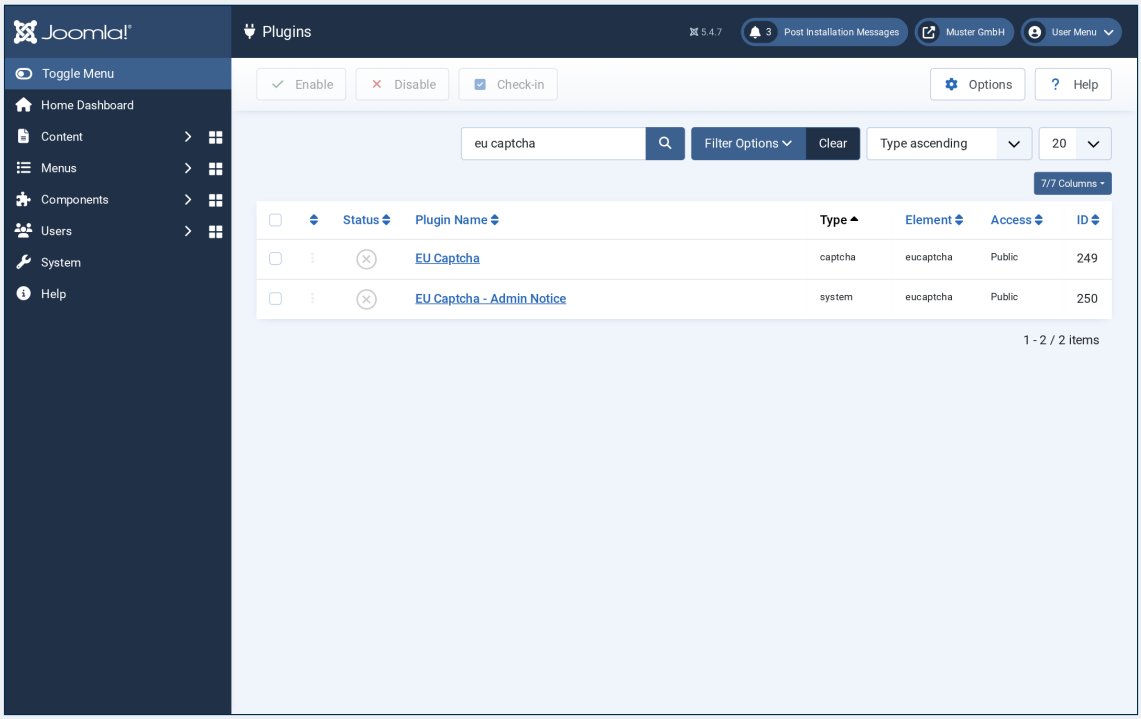


Abbildung 131: Liste der Plugins mit beiden Erweiterungen

- ► Markieren Sie beide Plugins und klicken Sie auf die Schaltfläche **Enable**.

➔ Beide Plugins sind eingeschaltet.

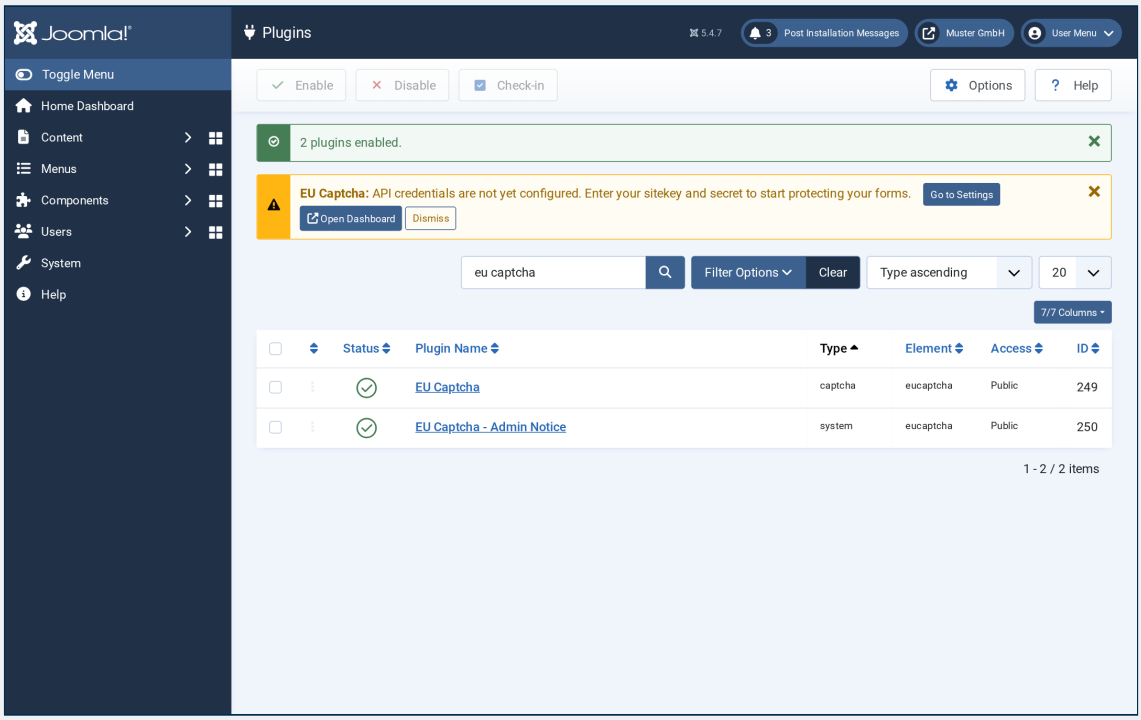


Abbildung 132: Liste der Plugins nach dem Einschalten

5.2.2.3 Zugangsdaten hinterlegen

Bis zur vollständigen Konfiguration zeigt Joomla auf jeder Seite des Administrationsbereichs einen Hinweis an.

Um die Zugangsdaten zu hinterlegen, gehen Sie wie folgt vor:

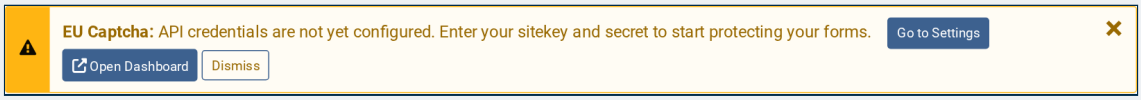


Abbildung 133: Hinweis im Administrationsbereich

► Klicken Sie im Hinweis auf die Schaltfläche **Go to Settings**. Alternativ öffnen Sie **Extensions** → **Plugins** und dort das Plugin **EU Captcha**.

↳ Die Einstellungen des Plugins werden angezeigt.

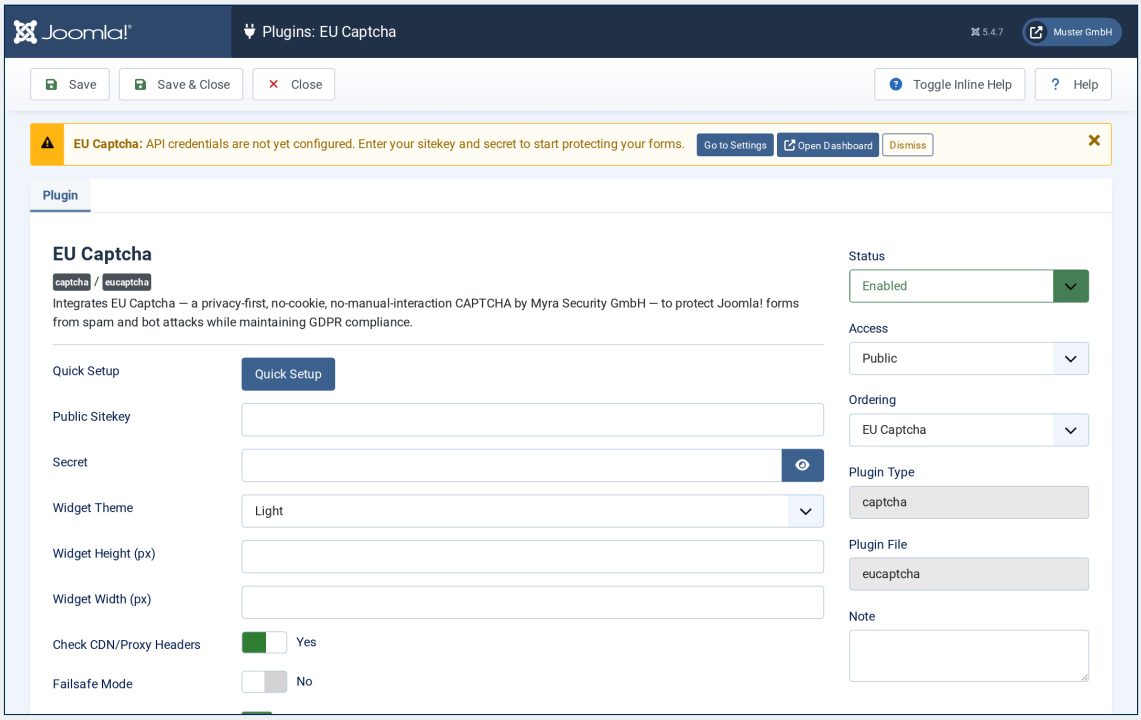


Abbildung 134: Einstellungen des Plugins EU Captcha

► Klicken Sie auf die Schaltfläche **Quick Setup**.

↳ Ein Dialog mit dem Registrierungsportal des Myra EU CAPTCHA wird angezeigt.

► Schließen Sie die Registrierung ab.

➔ Das Plugin übernimmt den öffentlichen Sitekey und das Secret.

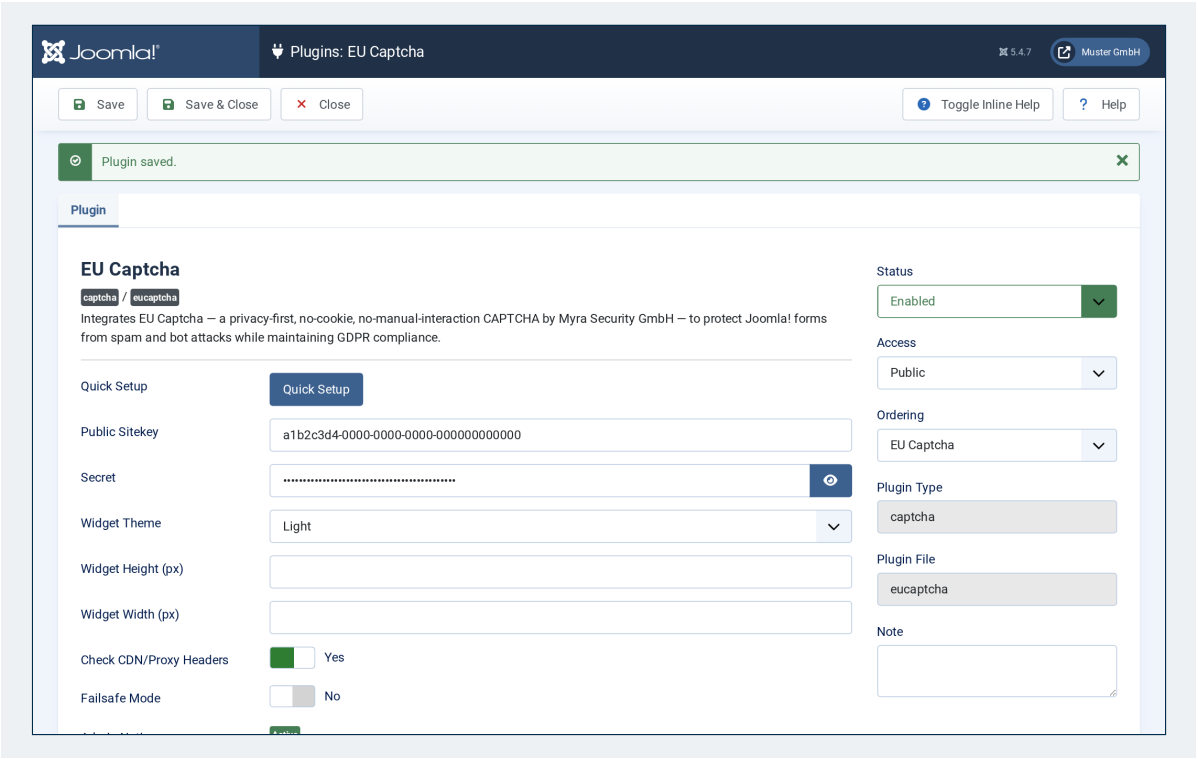


Abbildung 135: Einstellungen des Plugins nach dem Speichern

Alternativ tragen Sie vorhandene Zugangsdaten von Hand ein:

Feld	Inhalt
Public Sitekey	Öffentlicher Schlüssel im Format UUID. Der Wert darf im Frontend erscheinen.
Secret	Geheimer Schlüssel im Format Base64. Der Wert bleibt in der Datenbank von Joomla.

5.2.2.4 Optionen der API

Die folgenden Optionen stehen zur Verfügung:

Option	Vorgabe	Wirkung
Check CDN/ Proxy	**Yes**	Liest die tatsächliche IP-Adresse des Besuchers aus den Kopfzeilen `HTTP_CLIENT_IP`, `HTTP_X_FORWARDED_FOR` oder `HTTP_X_REAL_IP`,

Option	Vorgabe	Wirkung
Headers		bevor `REMOTE_ADDR` verwendet wird. Aktivieren Sie die Option, wenn Ihre Website hinter einem CDN oder einem Lastverteiler steht.
Failsafe Mode	**No**	Auf **No** blockiert ein Netzwerkfehler die Absendung. Auf **Yes** gilt ein Übertragungsfehler als bestandene Prüfung. Ungültige Token weist das Plugin in beiden Fällen zurück.

5.2.2.5 Darstellung des Widgets

Die folgenden Optionen steuern die Darstellung:

Option	Vorgabe	Wirkung
Widget Theme	**Light**	Helle oder dunkle Darstellung, Werte **Light** und **Dark**.
Widget Height (px)	leer	Feste Höhe in Pixeln, mindestens 48.
Widget Width (px)	leer	Feste Breite in Pixeln, mindestens 1.

5.2.2.6 CAPTCHA aktivieren

Um das CAPTCHA zu aktivieren, gehen Sie wie folgt vor:

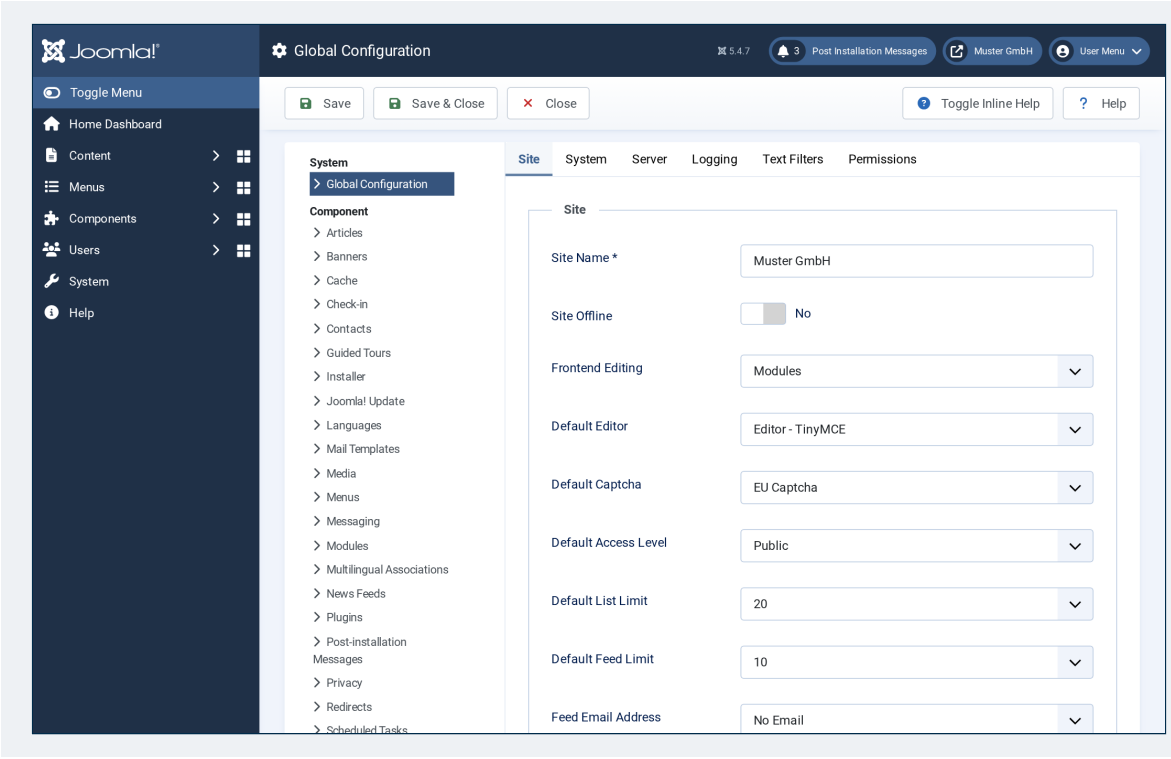


Abbildung 136: Ansicht Global Configuration mit der Option Default Captcha

- ► Öffnen Sie **System** → **Global Configuration** und wechseln Sie in den Reiter **Site**.
- ► Setzen Sie die Option **Default Captcha** auf den Wert **EU Captcha**.
- ► Klicken Sie auf die Schaltfläche **Save**.
- ➔ Alle Formulare mit einem CAPTCHA-Feld sind geschützt.

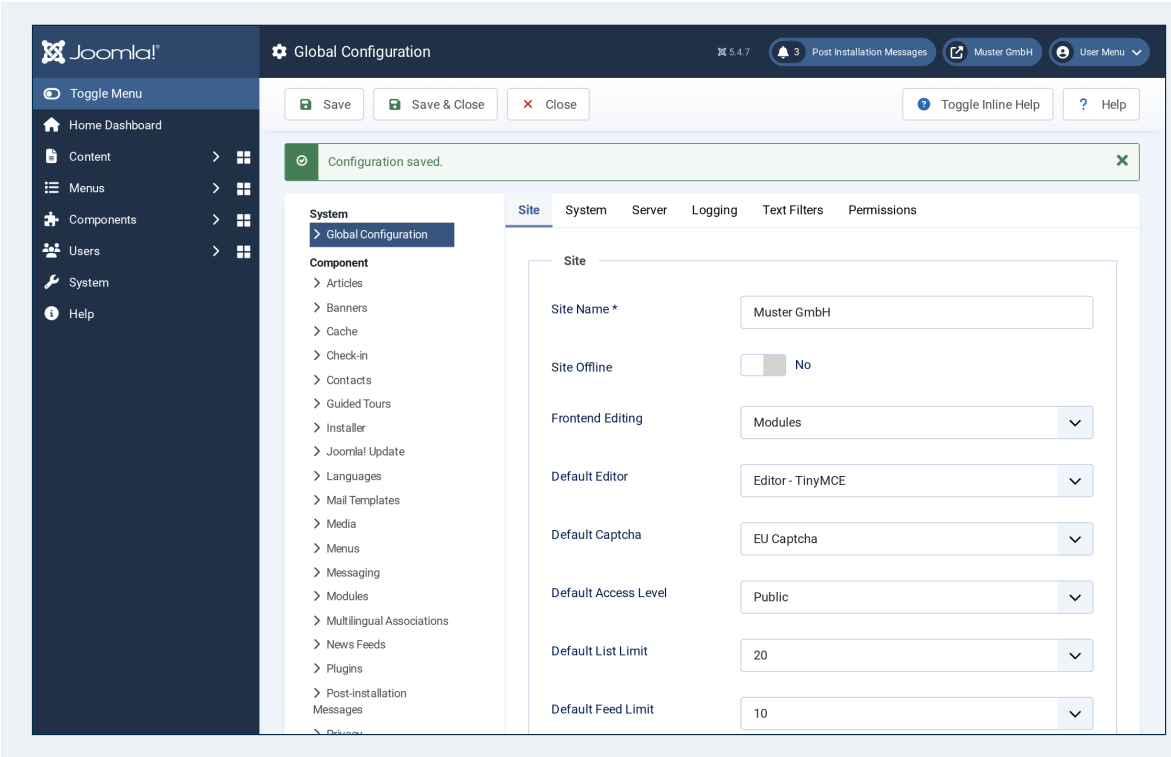


Abbildung 137: Ansicht Global Configuration nach dem Speichern

Für das mitgelieferte Kontaktformular aktivieren Sie das CAPTCHA-Feld zusätzlich unter **Components** → **Contact** → **Options** im Reiter **Form**.

5.2.2.7 Geschützte Formulare

Geschützt ist jedes Formular mit einem Standard-CAPTCHA-Feld, darunter:

- Registrierung von Nutzenden
- Zurücksetzen des Passworts
- Kontaktkomponente `com_contact`
- Komponenten von Drittanbietern mit Standard-CAPTCHA-Feld

5.2.3 Drupal

Das Modul für Drupal schützt die Formulare des Kerns und die Formulare des Moduls Webform. Das Widget wird aus dem CDN geladen. Eigener JavaScript-Quelltext ist nicht erforderlich.

5.2.3.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Drupal	Version 10 oder 11
Berechtigung	`administer eu captcha`
Zugang	Kommandozeile mit Drush

5.2.3.2 Modul installieren

Um das Modul zu installieren, gehen Sie wie folgt vor:

- Legen Sie das Modul im Verzeichnis `web/modules/custom/eu_captcha/` ab.
- Aktivieren Sie das Modul auf der Kommandozeile:

```
drush en eu_captcha
```

- Öffnen Sie im Administrationsbereich den Eintrag **Administration** → **Configuration** → **System** → **EU-Captcha**.

→ Die Einstellungsseite `/admin/config/system/eu-captcha` wird angezeigt.

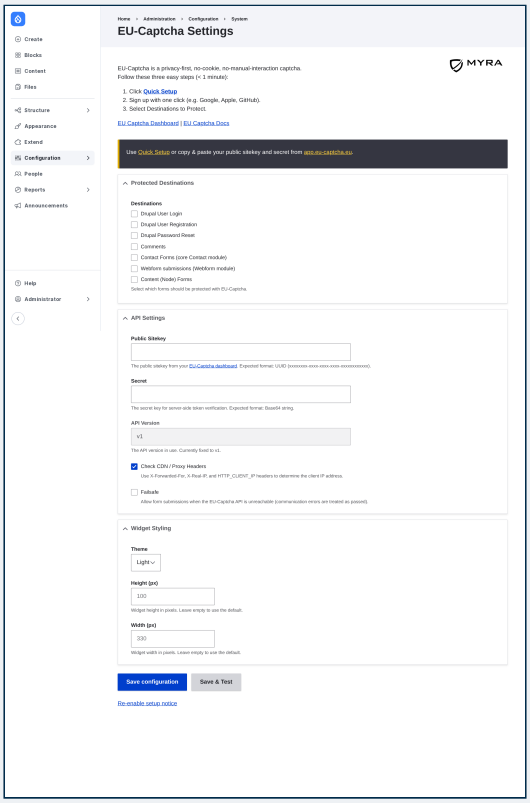


Abbildung 138: Einstellungsseite EU-Captcha Settings

5.2.3.3 Zugangsdaten hinterlegen

Um die Zugangsdaten zu hinterlegen, gehen Sie wie folgt vor:

- ► Klicken Sie auf der Einstellungsseite auf die Schaltfläche **Quick Setup**.
  - ↳ Ein Dialog mit der Anmeldung des Myra EU CAPTCHA wird angezeigt.
- ► Melden Sie sich mit Google, Apple oder GitHub an.
- ➔ Das Modul übernimmt den öffentlichen Sitekey und das Secret.

Alternativ tragen Sie vorhandene Zugangsdaten von Hand ein:

Feld	Inhalt
Public Sitekey	Öffentlicher Schlüssel im Format UUID. Das Widget im Frontend verwendet diesen Wert.
Secret	Geheimer Schlüssel im Format Base64 für die serverseitige Prüfung. Nach dem Speichern zeigt Drupal den Wert nicht mehr an.

Das Modul prüft beide Werte unmittelbar bei der Eingabe auf ihr Format.

5.2.3.4 Optionen der API

Die folgenden Optionen stehen zur Verfügung:

Option	Wirkung
Check CDN / Proxy Headers	Liest die tatsächliche IP-Adresse des Besuchers aus den Kopfzeilen `X-Forwarded-For`, `X-Real-IP` oder `HTTP_CLIENT_IP`. Aktivieren Sie die Option, wenn Drupal hinter einem Lastverteiler oder einem CDN steht.
Failsafe	Auf **On** lässt das Modul eine Absendung zu, wenn die API des Myra EU CAPTCHA nicht erreichbar ist. Auf **Off** blockiert das Modul die Absendung bei einem API-Fehler.

5.2.3.5 Formulare auswählen

Um die geschützten Formulare auszuwählen, gehen Sie wie folgt vor:

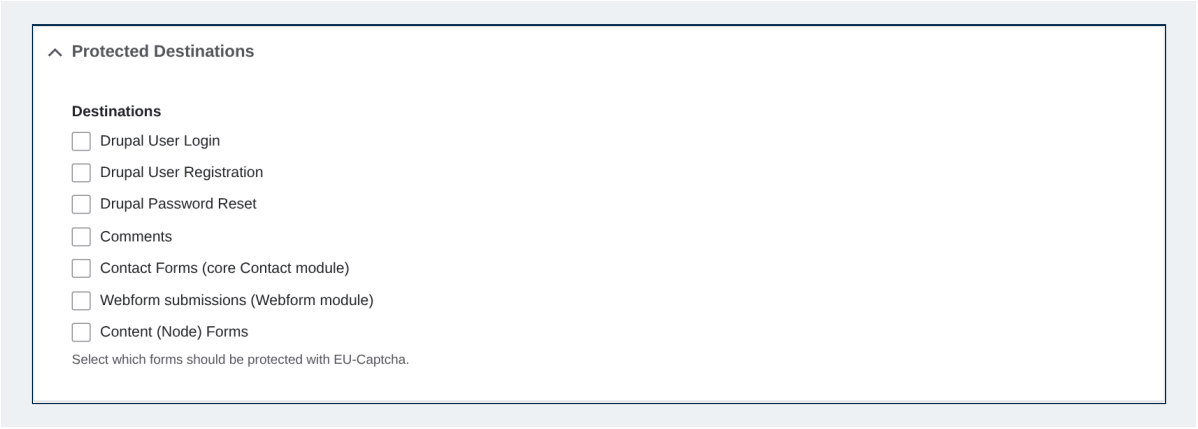


Abbildung 139: Bereich Protected Destinations

- ▶ Markieren Sie im Bereich **Protected Destinations** die gewünschten Ziele.
- ▶ Klicken Sie auf die Schaltfläche **Save configuration**.
- ➔ Das Widget erscheint in den markierten Formularen.

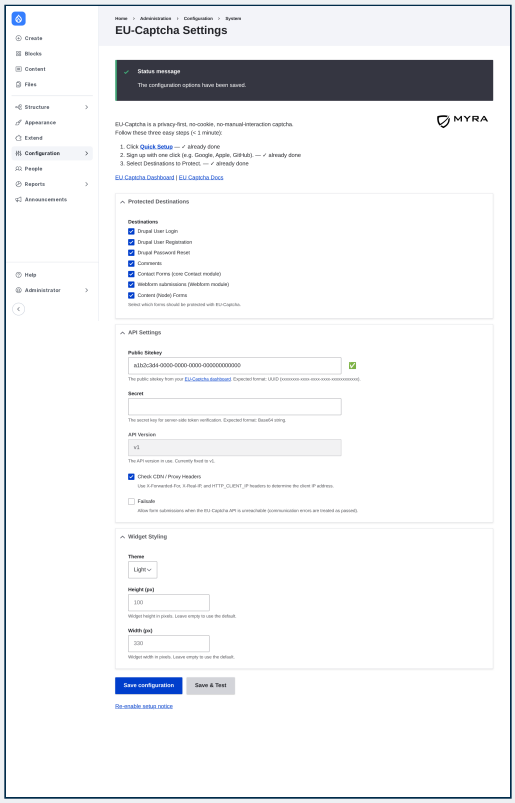


Abbildung 140: Einstellungsseite nach dem Speichern

Die folgenden Ziele stehen zur Verfügung:

Ziel	Beschriftung	Umfang
`user_login`	**Drupal User Login**	Anmeldeformular von Drupal
`user_register`	**Drupal User Registration**	Registrierungsformular von Drupal
`user_pass`	**Drupal Password Reset**	Formular zum Zurücksetzen des Passworts
`comment`	**Comments**	Alle Kommentarformulare (`comment_{bundle}_form`)
`contact`	**Contact Forms (core Contact module)**	Formulare des Kernmoduls Contact (`contact_message_{id}_form`)
`webform`		Formulare des Moduls Webform (`webform_submission_{id}_add_form`)

Ziel	Beschriftung	Umfang
	Webform submissions (Webform module)	
`node`	**Content (Node) Forms**	Formulare für Inhalte (`node_{typ}_form`)

5.2.3.6 Berechtigung

Berechtigung	Wirkung
`administer eu captcha`	Zugriff auf die Einstellungsseite, auf **Quick Setup** und auf den Hinweisbalken im Administrationsbereich. Vergeben Sie die Berechtigung nur an vertrauenswürdige Rollen.

5.2.3.7 Hinweisbalken

Solange kein Sitekey gespeichert oder kein Ziel markiert ist, zeigt Drupal im Administrationsbereich einen Hinweisbalken an. Der Balken lässt sich je Benutzer über die Schaltfläche **Dismiss** ausblenden und verschwindet dauerhaft, sobald ein Sitekey und mindestens ein Ziel gespeichert sind.

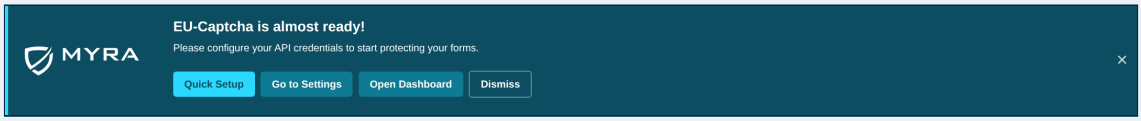


Abbildung 141: Hinweisbalken im Administrationsbereich

Um den Hinweisbalken erneut einzublenden, klicken Sie am Ende der Einstellungsseite auf den Link **Re-enable setup notice**.

5.2.3.8 Ablauf der Prüfung

1. Das Modul lädt `verify.js` aus dem CDN und fügt in jedes geschützte Formular ein Element `<div data-sitekey="…">` ein.
2. Beim Absenden ergänzt das Widget das verborgene Feld `eu-captcha-response` .

3. Der Server sendet Token, Sitekey, Secret und die IP-Adresse des Besuchers an `https://api.eu-captcha.eu/v1/verify`. Die Antwort `{"success": true}` lässt die Absendung zu. Jede andere Antwort erzeugt einen Validierungsfehler im Formular.

   Siehe **Widget wird nicht erkannt**.

5.2.4 TYPO3

Die Erweiterung für TYPO3 schützt die Anmeldung im Backend, die Anmeldung im Frontend über `EXT:felogin` und die Formulare des Formular-Frameworks `EXT:form`.

5.2.4.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
TYPO3	Version 14
Erweiterungen	`felogin` für die Anmeldung im Frontend, `form` für das Formular-Framework
Berechtigung	Administrationsrechte in TYPO3

5.2.4.2 Erweiterung installieren

Um die Erweiterung zu installieren, gehen Sie wie folgt vor:

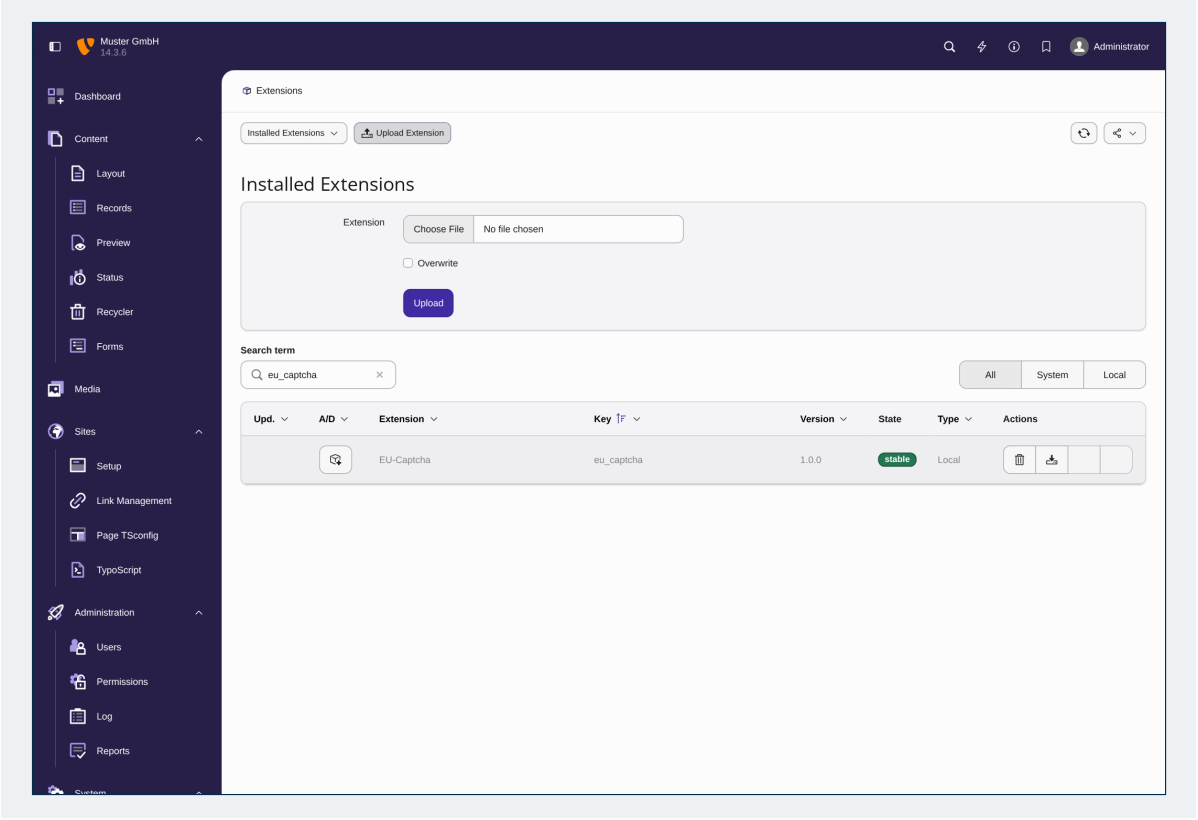


Abbildung 142: Modul Extensions mit dem geöffneten Formular zum Hochladen

- ► Klicken Sie in der Ansicht **Integration** auf die Schaltfläche **Download zip**.
- ► Öffnen Sie in TYPO3 das Modul **System → Extensions**.
- ► Klicken Sie auf die Schaltfläche **Upload Extension**.
- ► Wählen Sie das Archiv aus und klicken Sie auf die Schaltfläche **Upload**.

  ↳ TYPO3 meldet **Extension Upload** und führt die Erweiterung in der Liste **Installed Extensions**. Die Erweiterung ist noch nicht eingeschaltet.

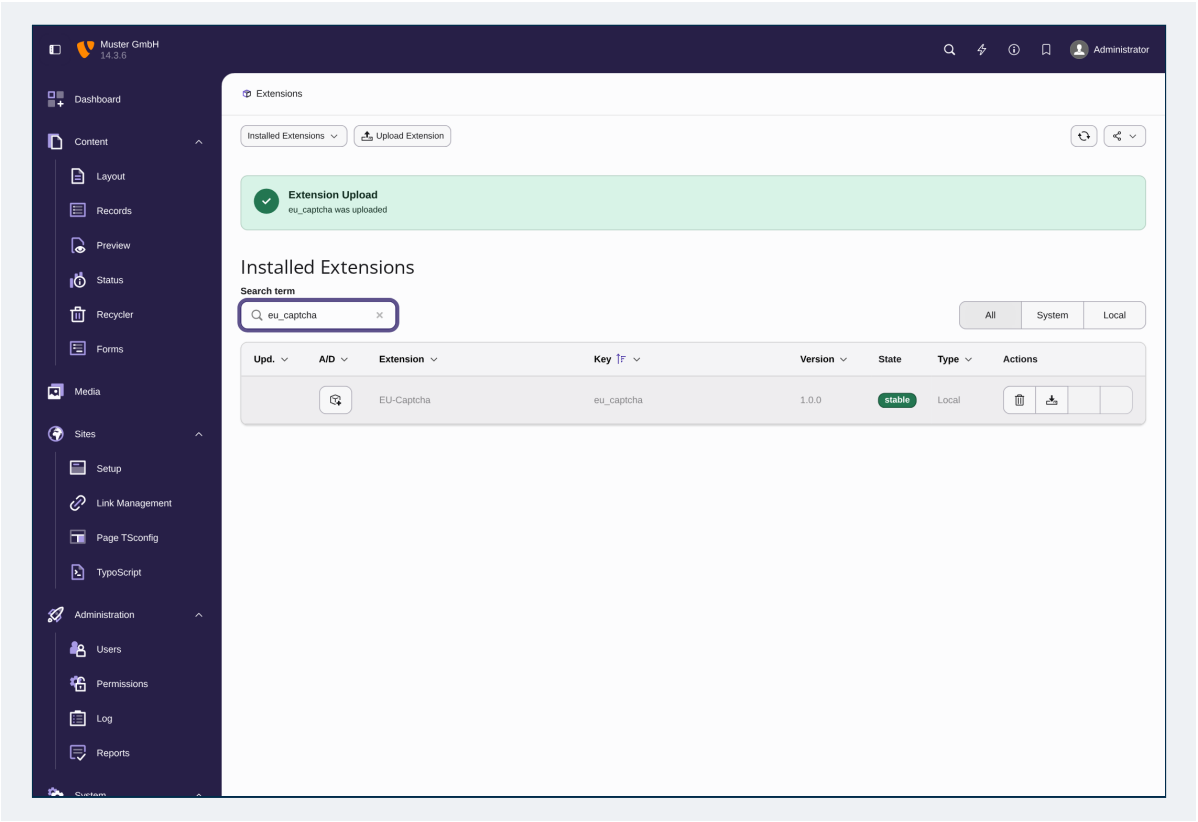


Abbildung 143: Liste Installed Extensions nach dem Hochladen

- ► Klicken Sie in der Spalte **A/D** auf das Symbol neben der Erweiterung **EU-Captcha**.
- ► Bestätigen Sie die Rückfrage.
  - ↳ TYPO3 lädt das Backend neu.
- ➔ Die Erweiterung ist eingeschaltet.

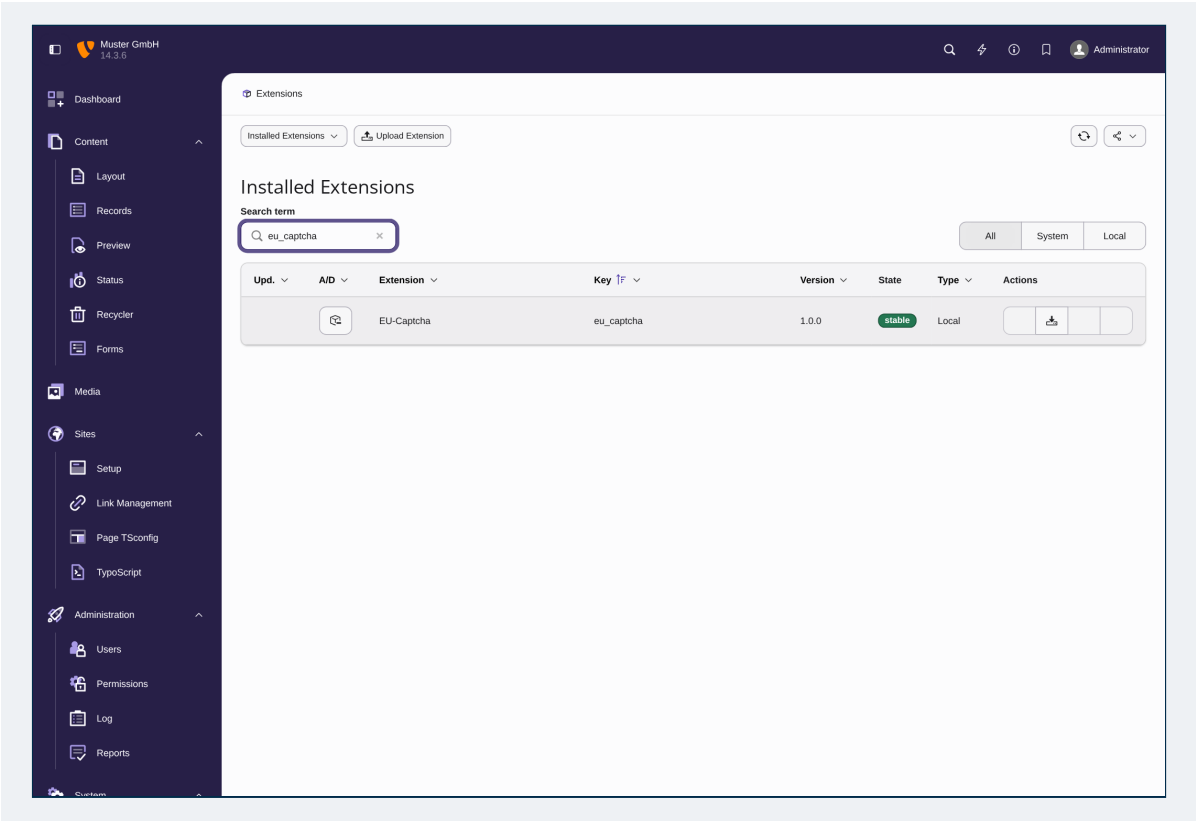


Abbildung 144: Liste Installed Extensions mit der eingeschalteten Erweiterung

5.2.4.3 TypoScript einbinden

Das TypoScript lädt das Skript des Widgets auf jeder Seite im Frontend und fügt das Widget in das Anmeldeformular von `felogin` ein. Ergänzen Sie die TypoScript-Datei Ihrer Website, zum Beispiel `typo3conf/sites/main/setup.typoscript` , um die folgende Zeile:

```
@import 'EXT:eu_captcha/Configuration/TypoScript/setup.typoscript'
```

Ab TYPO3 14 stammt das TypoScript einer Website aus einem Site-Set. Das Modul **Sites → TypoScript** zeigt die Definitionen nur an und weist darauf hin, dass sie im Dateisystem geändert werden.

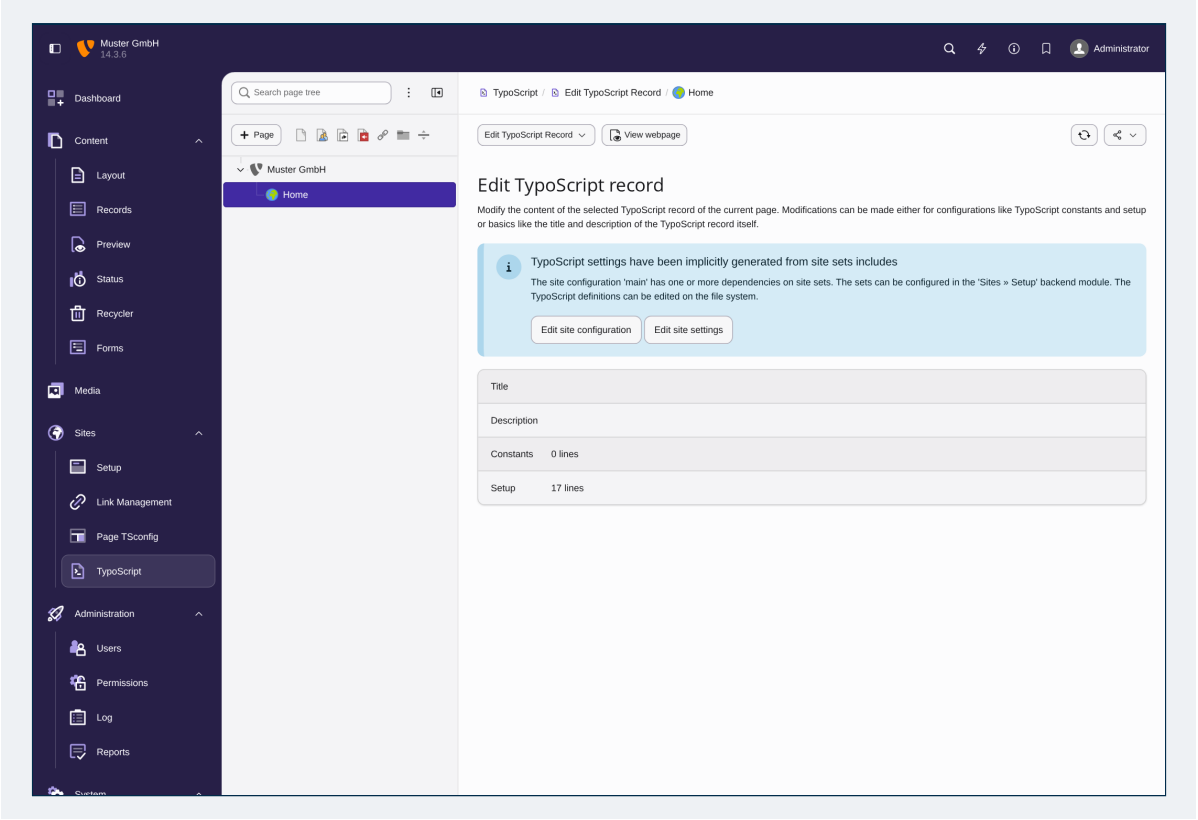


Abbildung 145: Modul TypoScript mit dem Hinweis auf das Site-Set

5.2.4.4 Zugangsdaten hinterlegen

Um die Zugangsdaten zu hinterlegen, gehen Sie wie folgt vor:

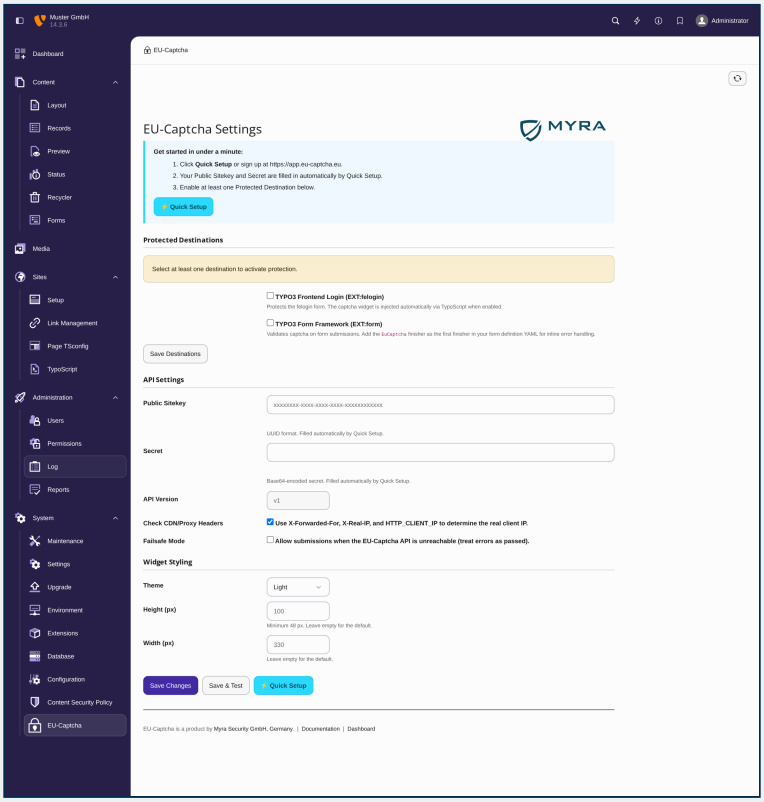


Abbildung 146: Modul EU-Captcha ohne Zugangsdaten

- ▶ Öffnen Sie in TYPO3 das Modul **System → EU-Captcha**.
- ▶ Klicken Sie auf die Schaltfläche **Quick Setup** und melden Sie sich an. Alternativ tragen Sie den öffentlichen Sitekey im Feld **Public Sitekey** und das Secret im Feld **Secret** von Hand ein. Beide Werte finden Sie in der Ansicht **Details** des Sitekeys.
- ▶ Klicken Sie auf die Schaltfläche **Save Changes**.
- ➔ Die Erweiterung speichert die Zugangsdaten.

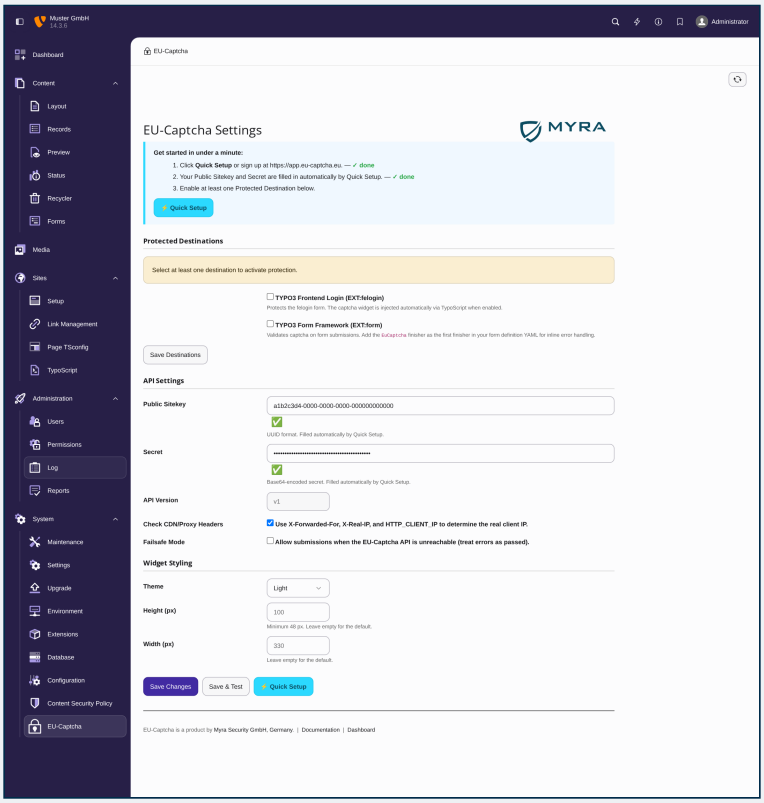


Abbildung 147: Modul EU-Captcha mit den gespeicherten Zugangsdaten

Die Schaltfläche **Save & Test** speichert die Zugangsdaten und prüft sie zusätzlich gegen die API.

Die Anmeldung im Backend von TYPO3 ist geschützt, sobald Sitekey und Secret gespeichert sind. Diese Prüfung lässt sich nicht abschalten.

5.2.4.5 Formulare auswählen

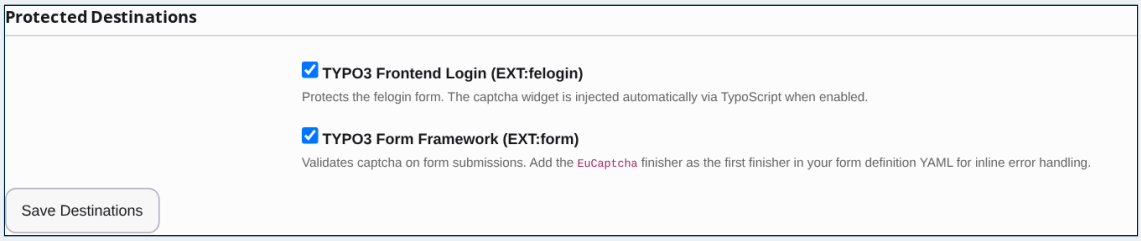


Abbildung 148: Bereich Protected Destinations mit beiden ausgewählten Zielen

Im Bereich **Protected Destinations** wählen Sie die geschützten Formulare aus. Solange kein Ziel markiert ist, zeigt TYPO3 den Hinweis `Select at least one destination to activate protection.` an.

Option	Umfang
TYPO3 Frontend Login (EXT:felogin)	Schützt das Anmeldeformular von `felogin`. Das Widget wird über das TypoScript automatisch eingefügt.
TYPO3 Form Framework (EXT:form)	Prüft die Absendung von Formularen aus `EXT:form`. Tragen Sie in der YAML-Definition des Formulars den Finisher `EuCaptcha` als ersten Finisher ein, damit Fehler unmittelbar im Formular erscheinen.

Klicken Sie anschließend auf die Schaltfläche **Save Destinations**.

5.2.4.6 Optionen der API

Die folgenden Optionen stehen zur Verfügung:

Option	Wirkung
Check CDN/ Proxy Headers	Liest die tatsächliche IP-Adresse des Besuchers aus den Kopfzeilen des vorgelagerten Systems. Aktivieren Sie die Option, wenn TYPO3 hinter einem CDN oder einem Lastverteiler steht.
Failsafe Mode	Lässt eine Absendung zu, wenn die API des Myra EU CAPTCHA nicht erreichbar ist.

5.2.4.7 Darstellung des Widgets

Die folgenden Optionen steuern die Darstellung:

Option	Vorgabe	Wirkung
Theme	**Light**	Helle oder dunkle Darstellung, Werte **Light** und **Dark**.
	leer	

Option	Vorgabe	Wirkung
Height (px)		Feste Höhe in Pixeln, mindestens 48. Bei einem leeren Feld gilt der Standardwert.
Width (px)	leer	Feste Breite in Pixeln, mindestens 1. Bei einem leeren Feld gilt der Standardwert.

Klicken Sie auf die Schaltfläche **Save Changes**, um die Darstellung zu speichern.

5.2.4.8 Content Security Policy

Die Erweiterung liefert eine eigene Content Security Policy mit. Diese Richtlinie gibt für das Backend und für das Frontend die Direktive `script-src` für die Adresse `https://cdn.eu-captcha.eu` frei.

Die Direktiven `frame-src` und `connect-src` enthält die mitgelieferte Richtlinie nur für das Backend. Wenn Sie im Frontend eine Content Security Policy verwenden, dann ergänzen Sie diese beiden Direktiven selbst. Ohne diese Ergänzung lädt das Skript, das versteckte Iframe bleibt jedoch leer.

Siehe **Content Security Policy**.

5.2.4.9 Integration prüfen

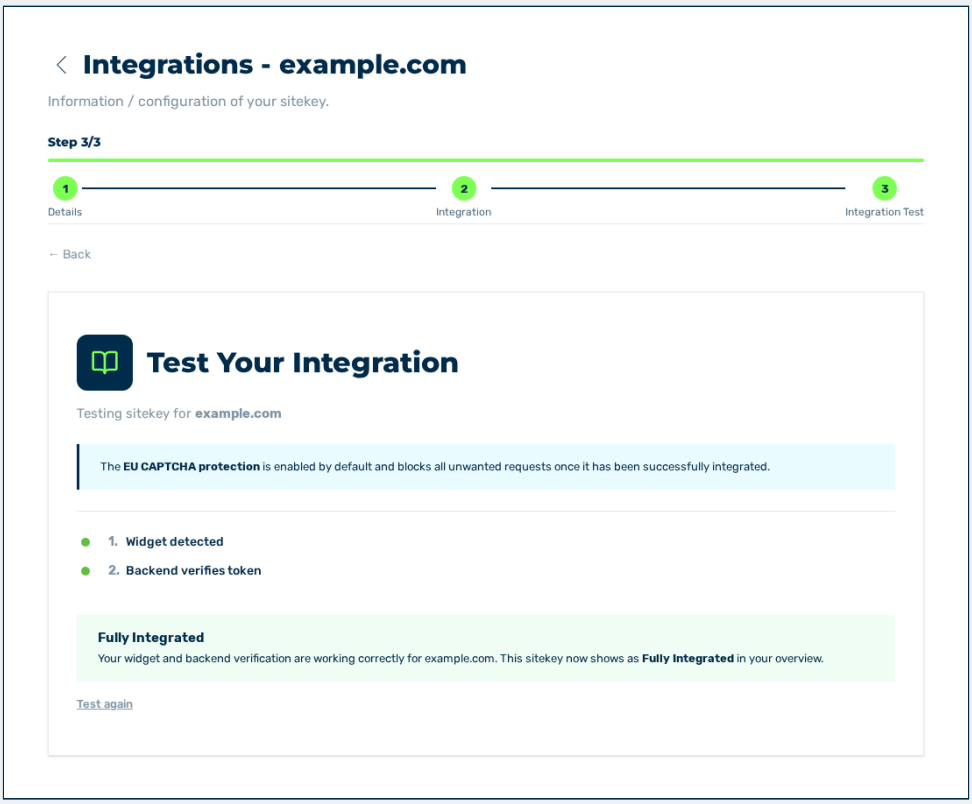


Abbildung 149: Ansicht Integration Test mit dem Ergebnis Fully Integrated

Wechseln Sie nach der Einrichtung in die Ansicht **Integration Test** des Sitekeys und prüfen Sie die Einbindung.

Siehe **Integration testen**.

5.3 Frontend

5.3.1 HTML und JavaScript

Ohne Framework binden Sie das Widget mit zwei Zeilen HTML ein: einem Skript-Verweis im Kopfbereich der Seite und einem Element im Formular. Diesen Weg verwenden Sie auch für jedes Content-Management-System, das eigenes HTML zulässt.

5.3.1.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Sitekey	Öffentlicher Sitekey aus der Ansicht **Details**
Zugriff	Schreibrechte auf die HTML-Vorlage der Seite
Serverseite	Endpunkt, der das Token prüft

5.3.1.2 Skript einbinden

Ergänzen Sie den Bereich `<head>` Ihrer Seite um die folgende Zeile:

```
<script src="https://cdn.eu-captcha.eu/verify.js" async defer></script>
```

Wenn Ihre Website eine Content Security Policy sendet, dann muss diese die Adressen `https://cdn.eu-captcha.eu` und `https://api.eu-captcha.eu` freigeben. Siehe **Content Security Policy**.

5.3.1.3 Widget einfügen

Fügen Sie in jedes zu schützende Formular das folgende Element ein:

```
<div class="eu-captcha" data-sitekey="a1b2c3d4-0000-0000-0000-000000000000"></
div>
```

Ersetzen Sie den Wert von `data-sitekey` durch Ihren öffentlichen Sitekey.

Das vollständige Formular sieht damit wie folgt aus:

```
<form method="POST" action="/your-endpoint">
  <!-- your fields -->

  <div class="eu-captcha" data-
sitekey="a1b2c3d4-0000-0000-0000-000000000000"></div>

  <button type="submit">Submit</button>
</form>
```

Beim Absenden ergänzt das Widget das verborgene Feld `eu-captcha-response` mit dem Token.

5.3.1.4 Attribute des Elements

Das Skript `verify.js` wertet die folgenden Attribute aus:

Attribut	Vorgabe	Wirkung
`data-sitekey`	—	Öffentlicher Sitekey. Der Wert ist erforderlich.
`data-theme`	`"light"`	Darstellung, Werte `"light"` und `"dark"`.
`data-width`	`330`	Breite des Widgets in Pixeln.
`data-height`	`100`	Höhe des Widgets in Pixeln.
`data-widgetid`	automatisch	Eigene Kennung des Widgets.
`data-autostart`	`"true"`	Der Wert `"false"` verschiebt den Start der Challenge.
`data-callback`	—	Name einer globalen Funktion, die nach der bestandenen Challenge aufgerufen wird.
`data-expired-callback`	—	Name einer globalen Funktion, die beim Ablauf des Tokens aufgerufen wird.
	—	

Attribut	Vorgabe	Wirkung
`data-error-callback`		Name einer globalen Funktion, die bei einem Fehler aufgerufen wird.

5.3.1.5 Paket über npm einbinden

Für Projekte mit einem Bündelwerkzeug steht dasselbe Widget als npm-Paket bereit. Das Paket liefert TypeScript-Typen und eine Versionierung. Das Widget lädt weiterhin `verify.js` aus dem CDN.

Um das Paket einzubinden, gehen Sie wie folgt vor:

- ▸ Installieren Sie das Paket:

```
npm i @myrasec/eu-captcha-vanilla
```

- ▸ Legen Sie im Formular ein Zielelement an:

```
<form id="contact-form">
  <!-- your fields -->
  <div id="captcha"></div>
  <button type="submit">Submit</button>
</form>
```

- ▸ Zeichnen Sie das Widget in das Zielelement:

```
import { renderEuCaptcha, isEuCaptchaDone } from "@myrasec/eu-captcha-
vanilla";

const captchaSitekey = "EUCAPTCHA_SITE_KEY";

renderEuCaptcha("#captcha", {
  sitekey: captchaSitekey,
  onComplete: (token: string) => console.log("token:", token),
}).catch((err) => console.error("EU CAPTCHA failed to render", err));
```

- ➔ Das Widget beginnt die Prüfung im Hintergrund.

Als erstes Argument nimmt `renderEuCaptcha` entweder einen CSS-Selektor oder ein `HTMLElement` entgegen.

5.3.1.6 Optionen

Die folgenden Optionen stehen zur Verfügung:

Option	Typ	Vorgabe	Wirkung
`sitekey`	`string`	—	Öffentlicher Sitekey. Der Wert ist erforderlich.
`theme`	`string`	`"light"`	Darstellung, Werte `"light"` und `"dark"`.
`width`	`number`	`330`	Breite des Widgets in Pixeln.
`height`	`number`	`100`	Höhe des Widgets in Pixeln.
`widgetId`	`string`	—	Eigene Kennung des Widgets. Ohne Angabe erzeugt das Paket eine Kennung. Für `euCaptcha.execute()` ist der Wert erforderlich.
`autostart`	`boolean`	`true`	Startet die Challenge automatisch. Der Wert `false` verschiebt den Start auf den Aufruf `euCaptcha.execute(widgetId)`.
`onComplete`	`(token: string) => void`	—	Wird mit dem Token aufgerufen, sobald die Challenge bestanden ist.
`onExpired`	`() => void`	—	Wird aufgerufen, wenn das Token abläuft. Ein Token läuft 60 Minuten nach der Lösung ab.
`onError`	`() => void`	—	Wird bei einem Netzwerk- oder Serverfehler aufgerufen.

5.3.1.7 Challenge verzögert starten

Setzen Sie `autostart` auf `false` und vergeben Sie eine Kennung, um die Challenge selbst zu starten:

```
renderEuCaptcha("#captcha", {
  sitekey: captchaSitekey,
  widgetId: "my-captcha",
  autostart: false,
});

document.getElementById("verify-btn")!.addEventListener("click", () => {
  (window as any).euCaptcha.execute("my-captcha");
});
```

5.3.1.8 Zustand abfragen

Prüfen Sie vor dem Absenden, ob die Challenge bestanden ist:

```
import { isEuCaptchaDone } from "@myrasec/eu-captcha-vanilla";

function handleSubmit(e: SubmitEvent): void {
    e.preventDefault();

    if (!isEuCaptchaDone()) {
        // challenge not yet complete
        return;
    }
    // proceed with form submission
}
```

Ohne Argument ist `isEuCaptchaDone()` nur bei genau einem Widget ohne eigene Kennung zuverlässig. Vergeben Sie bei mehreren Widgets je Widget eine Kennung und fragen Sie `isEuCaptchaDone(widgetId)` ab.

Alternativ warten Sie auf die Nachricht `euCaptchaCompleted` am Fenster. Prüfen Sie dabei den Ursprung der Nachricht, da jedes Skript und jede Erweiterung im Browser Nachrichten senden kann:

```
const CAPTCHA_ORIGIN = "https://cdn.eu-captcha.eu";

function listenForCaptchaDone(msg: MessageEvent): void {
```

```
    if (msg.origin !== CAPTCHA_ORIGIN) return;
    const data = (msg.data ?? {}) as { type?: string };
    if (data.type === "euCaptchaCompleted") {
        // enable submit button, update state, etc.
    }
}

window.addEventListener("message", listenForCaptchaDone, false);
```

Die Prüfung im Browser steuert nur die Bedienung. Prüfen Sie das Token in jedem Fall zusätzlich auf dem Server.

5.3.1.9 Widget abbauen

`renderEuCaptcha` gibt eine Referenz zurück. Der Aufruf `handle.destroy()` entfernt den Nachrichtenempfänger, leert das Zielelement und setzt den Zustand zurück, sodass `isEuCaptchaDone()` wieder `false` meldet.

```
const handle = await renderEuCaptcha("#captcha", { sitekey: captchaSitekey });
// later
handle.destroy();
```

Rufen Sie `destroy()` bei jedem Abbau auf. Jede Darstellung mit einer Rückruffunktion fügt dem Fenster einen Nachrichtenempfänger hinzu, den nur `destroy()` wieder entfernt. Ohne diesen Aufruf bleibt je Darstellung ein Empfänger und ein abgetrennter DOM-Teilbaum im Speicher zurück.

5.3.1.10 Token prüfen

Prüfen Sie das Token auf Ihrem Server. Siehe **Endpunkt /verify**.

5.3.1.11 Vollständiges Beispiel

Siehe **HTML und Django**.

5.3.2 React

Das Paket `@myrasec/eu-captcha` stellt die Komponente `EuCaptcha` bereit. Die Komponente zeichnet das Widget im Formular und meldet das Token an Ihren Quelltext.

5.3.2.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
React	ab Version 18
Sitekey	Öffentlicher Sitekey aus der Ansicht **Details**
Serverseite	Endpunkt, der das Token prüft

5.3.2.2 Paket installieren

Installieren Sie das Paket:

```
npm i @myrasec/eu-captcha
```

5.3.2.3 Komponente einbinden

Binden Sie die Komponente in das zu schützende Formular ein:

```
import { EuCaptcha, isEuCaptchaDone } from "@myrasec/eu-captcha";

const captchaSitekey = "EUCAPTCHA_SITE_KEY";

export function ContactForm() {
  function handleSubmit(e: React.FormEvent<HTMLFormElement>): void {
    e.preventDefault();
    if (!isEuCaptchaDone()) {
      // challenge not yet complete
      return;
    }
    // proceed with form submission
  }

  return (
    <form onSubmit={handleSubmit}>
      {/* your fields */}
```

```
        <EuCaptcha sitekey={captchaSitekey} />
        <button type="submit">Submit</button>
      </form>
    );
  }
```

> Sie prüfen die Einbindung mit einem beliebigen erfundenen Sitekey. Zu einem unbekannten Sitekey läuft das CAPTCHA mit den Vorgabewerten, und der gesamte Verkehr wird durchgelassen.

5.3.2.4 Eigenschaften

Die folgenden Eigenschaften stehen zur Verfügung:

Eigenschaft	Typ	Vorgabe	Wirkung
`sitekey`	`string`	—	Öffentlicher Sitekey. Der Wert ist erforderlich.
`theme`	`string`	`"light"`	Darstellung, Werte `"light"` und `"dark"`.
`width`	`number`	`330`	Breite des Widgets in Pixeln.
`height`	`number`	`100`	Höhe des Widgets in Pixeln.
`widgetId`	`string`	—	Eigene Kennung des Widgets. Ohne Angabe erzeugt das Paket eine Kennung. Für `euCaptcha.execute()` ist der Wert erforderlich.
`autostart`	`boolean`	`true`	Startet die Challenge automatisch. Der Wert `false` verschiebt den Start auf den Aufruf `euCaptcha.execute(widgetId)`.
`onComplete`	`(token: string) => void`	—	Wird mit dem Token aufgerufen, sobald die Challenge bestanden ist.

Eigenschaft	Typ	Vorgabe	Wirkung
`onExpired`	`() => void`	—	Wird aufgerufen, wenn das Token abläuft. Ein Token läuft 60 Minuten nach der Lösung ab.
`onError`	`() => void`	—	Wird bei einem Netzwerk- oder Serverfehler aufgerufen.

Beispiele für die Darstellung:

```
<EuCaptcha sitekey={captchaSitekey} theme="dark" />
<EuCaptcha sitekey={captchaSitekey} width={280} />
<EuCaptcha sitekey={captchaSitekey} height={60} />
```

5.3.2.5 Rückrufe verwenden

```
<EuCaptcha
  sitekey={captchaSitekey}
  onComplete={(token: string) => console.log("token:", token)}
  onExpired={() => console.log("token expired")}
  onError={() => console.log("challenge error")}
/>
```

5.3.2.6 Challenge verzögert starten

Setzen Sie `autostart` auf `false` und vergeben Sie eine Kennung:

```
<EuCaptcha
  sitekey={captchaSitekey}
  widgetId="my-captcha"
  autostart={false}
/>

<button onClick={() => (window as any).euCaptcha.execute("my-captcha")}
>Verify</button>
```

5.3.2.7 Next.js

Die Komponente greift auf Schnittstellen des Browsers zu und läuft daher nur im Browser.

Im **App Router** setzen Sie `'use client'` an den Anfang der Datei:

```
'use client';

import { EuCaptcha, isEuCaptchaDone } from "@myrasec/eu-captcha";
```

Im **Pages Router** laden Sie die Komponente über `next/dynamic` ohne serverseitige Darstellung:

```
import dynamic from "next/dynamic";
import { isEuCaptchaDone } from "@myrasec/eu-captcha";

const EuCaptcha = dynamic(
  () => import("@myrasec/eu-captcha").then((m) => m.EuCaptcha),
  { ssr: false }
);
```

Dasselbe Vorgehen gilt für Gatsby, da auch dort das Paket `@myrasec/eu-captcha` zum Einsatz kommt.

5.3.2.8 Paket ohne React verwenden

Das Paket lässt sich auch ohne React verwenden. React bleibt dabei eine Abhängigkeit des Pakets.

Um das Paket ohne React zu verwenden, gehen Sie wie folgt vor:

- Laden Sie die Bestandteile asynchron:

```
import { loadEuCaptcha, isEuCaptchaDone } from "@myrasec/eu-captcha";

loadEuCaptcha();
```

► Fügen Sie das Element in das Formular ein:

```
<div class="eu-captcha" data-sitekey="EUCAPTCHA_SITE_KEY"></div>
```

➔ Das Widget wird im Formular gezeichnet.

Die unterstützten Attribute sind unter **HTML und JavaScript** beschrieben.

5.3.2.9 Zustand abfragen

Prüfen Sie vor dem Absenden, ob die Challenge bestanden ist:

```
import { isEuCaptchaDone } from "@myrasec/eu-captcha";

function handleSubmit(e: React.FormEvent<HTMLFormElement>): void {
    e.preventDefault();

    if (!isEuCaptchaDone()) {
        // challenge not yet complete
        return;
    }
    // proceed with form submission
}
```

Alternativ warten Sie auf die Nachricht `euCaptchaDone` am Fenster:

```
function listenForCaptchaDone(msg: MessageEvent): void {
    if (msg.data.type === "euCaptchaDone") {
        // enable submit button, update state, etc.
    }
}

window.addEventListener("message", listenForCaptchaDone, false);
```

Die Prüfung im Browser steuert nur die Bedienung. Prüfen Sie das Token in jedem Fall zusätzlich auf dem Server. Siehe **Endpunkt /verify**.

5.3.2.10 Vollständiges Beispiel

Siehe **React und PHP**.

5.3.3 Vue

Das Paket `@myrasec/eu-captcha-vue` stellt die Komponente `EuCaptcha` für Vue 3 und Nuxt bereit.

5.3.3.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Vue	Version 3, auch als Grundlage von Nuxt
Sitekey	Öffentlicher Sitekey aus der Ansicht **Details**
Serverseite	Endpunkt, der das Token prüft

5.3.3.2 Paket installieren

Installieren Sie das Paket:

```
npm i @myrasec/eu-captcha-vue
```

5.3.3.3 Komponente einbinden

Binden Sie die Komponente in das zu schützende Formular ein:

```
<script setup lang="ts">
import { EuCaptcha, isEuCaptchaDone } from "@myrasec/eu-captcha-vue";

const captchaSitekey = "EUCAPTCHA_SITE_KEY";

function handleSubmit(): void {
  if (!isEuCaptchaDone()) {
    // challenge not yet complete
    return;
  }
  // proceed with form submission
}
</script>

<template>
  <form @submit.prevent="handleSubmit">
    <!-- your fields -->
```

```
    <EuCaptcha :sitekey="captchaSitekey" />
    <button type="submit">Submit</button>
  </form>
</template>
```

Sie prüfen die Einbindung mit einem beliebigen erfundenen Sitekey. Zu einem unbekannten Sitekey läuft das CAPTCHA mit den Vorgabewerten, und der gesamte Verkehr wird durchgelassen.

5.3.3.4 Eigenschaften

Die folgenden Eigenschaften stehen zur Verfügung:

Eigenschaft	Typ	Vorgabe	Wirkung
`sitekey`	`string`	—	Öffentlicher Sitekey. Der Wert ist erforderlich.
`theme`	`string`	`"light"`	Darstellung, Werte `"light"` und `"dark"`.
`width`	`number`	`330`	Breite des Widgets in Pixeln.
`height`	`number`	`100`	Höhe des Widgets in Pixeln.
`widgetId`	`string`	—	Eigene Kennung des Widgets. Ohne Angabe erzeugt das Paket eine Kennung. Für `euCaptcha.execute()` ist der Wert erforderlich.
`autostart`	`boolean`	`true`	Startet die Challenge beim Einhängen der Komponente. Der Wert `false` verschiebt den Start auf den Aufruf `euCaptcha.execute(widgetId)`.
`onComplete`	`(token: string) => void`	—	Wird mit dem Token aufgerufen, sobald die Challenge bestanden ist.

Eigenschaft	Typ	Vorgabe	Wirkung
`onExpired`	`() => void`	—	Wird aufgerufen, wenn das Token abläuft. Ein Token läuft 60 Minuten nach der Lösung ab.
`onError`	`() => void`	—	Wird bei einem Netzwerk- oder Serverfehler aufgerufen.

Beispiele für die Darstellung:

```
<EuCaptcha :sitekey="captchaSitekey" theme="dark" />
<EuCaptcha :sitekey="captchaSitekey" :width="280" />
<EuCaptcha :sitekey="captchaSitekey" :height="60" />
```

5.3.3.5 Rückrufe verwenden

```
<EuCaptcha
  :sitekey="captchaSitekey"
  :on-complete="(token) => console.log('token:', token)"
  :on-expired="() => console.log('token expired')"
  :on-error="() => console.log('challenge error')"
/>
```

5.3.3.6 Challenge verzögert starten

Setzen Sie `autostart` auf `false` und vergeben Sie eine Kennung:

```
<EuCaptcha
  :sitekey="captchaSitekey"
  widget-id="my-captcha"
  :autostart="false"
/>

<button @click="euCaptcha.execute('my-captcha')">Verify</button>
```

5.3.3.7 Zustand abfragen

Prüfen Sie vor dem Absenden, ob die Challenge bestanden ist:

```
import { isEuCaptchaDone } from "@myrasec/eu-captcha-vue";

function handleSubmit(): void {
    if (!isEuCaptchaDone()) {
        // challenge not yet complete
        return;
    }
    // proceed with form submission
}
```

Alternativ warten Sie auf die Nachricht `euCaptchaDone` am Fenster:

```
import { onMounted, onUnmounted } from "vue";

function listenForCaptchaDone(msg: MessageEvent): void {
    if (msg.data.type === "euCaptchaDone") {
        // enable submit button, update reactive state, etc.
    }
}

onMounted(() => window.addEventListener("message", listenForCaptchaDone,
false));
onUnmounted(() => window.removeEventListener("message", listenForCaptchaDone,
false));
```

Die Prüfung im Browser steuert nur die Bedienung. Prüfen Sie das Token in jedem Fall zusätzlich auf dem Server. Siehe **Endpunkt /verify**.

5.3.3.8 Vollständiges Beispiel

Siehe **Vue und Python**.

5.3.4 Angular

Für Angular steht je Hauptversion ein eigenes Paket bereit. Alle Pakete stellen die Komponente `eu-captcha` mit denselben Eingaben und Ausgaben bereit.

5.3.4.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Angular	Version 19, 20 oder 21
Sitekey	Öffentlicher Sitekey aus der Ansicht **Details**
Serverseite	Endpunkt, der das Token prüft

5.3.4.2 Paket wählen

Wählen Sie das Paket zur Hauptversion Ihres Projekts:

Paket	Angular
`@myrasec/eu-captcha-angular19`	Version 19
`@myrasec/eu-captcha-angular20`	Version 20
`@myrasec/eu-captcha-angular21`	Version 21

Installieren Sie das gewählte Paket, hier am Beispiel von Angular 21:

```
npm i @myrasec/eu-captcha-angular21
```

5.3.4.3 Komponente einbinden

In einer eigenständigen Komponente binden Sie `EuCaptchaComponent` ein:

```
import { Component } from '@angular/core';
import { EuCaptchaComponent, isEuCaptchaDone } from '@myrasec/eu-captcha-
angular21';

@Component({
```

```
  standalone: true,
  imports: [EuCaptchaComponent],
  template: `
    <form (submit)="handleSubmit($event)">
      <!-- your fields -->
      <eu-captcha sitekey="EUCAPTCHA_SITE_KEY" />
      <button type="submit">Submit</button>
    </form>
  `,
})
export class MyFormComponent {
  handleSubmit(event: Event): void {
    event.preventDefault();
    if (!isEuCaptchaDone()) {
      // challenge not yet complete
      return;
    }
    // proceed with form submission
  }
}
```

In einer Anwendung mit Modulen binden Sie stattdessen `EuCaptchaModule` ein:

```
import { NgModule } from '@angular/core';
import { EuCaptchaModule } from '@myrasec/eu-captcha-angular21';

@NgModule({
  imports: [EuCaptchaModule],
})
export class AppModule {}
```

```
<!-- in your template -->
<eu-captcha sitekey="EUCAPTCHA_SITE_KEY" />
```

Sie prüfen die Einbindung mit einem beliebigen erfundenen Sitekey. Zu einem unbekannten Sitekey läuft das CAPTCHA mit den Vorgabewerten, und der gesamte Verkehr wird durchgelassen.

5.3.4.4 Eingaben

Die folgenden Eingaben stehen zur Verfügung:

Eingabe	Typ	Vorgabe	Wirkung
`sitekey`	`string`	—	Öffentlicher Sitekey. Der Wert ist erforderlich.
`theme`	`string`	`"light"`	Darstellung, Werte `"light"` und `"dark"`.
`width`	`number`	`330`	Breite des Widgets in Pixeln.
`height`	`number`	`100`	Höhe des Widgets in Pixeln.
`widgetId`	`string`	—	Eigene Kennung des Widgets. Ohne Angabe erzeugt das Paket eine Kennung. Für `euCaptcha.execute()` ist der Wert erforderlich.
`autostart`	`boolean`	`true`	Startet die Challenge beim Einhängen der Komponente. Der Wert `false` verschiebt den Start auf den Aufruf `euCaptcha.execute(widgetId)`.

5.3.4.5 Ausgaben

Die folgenden Ausgaben stehen zur Verfügung:

Ausgabe	Wert	Wirkung
`completed`	`string`	Wird mit dem Token ausgelöst, sobald die Challenge bestanden ist.
`expired`	—	Wird ausgelöst, wenn das Token abläuft. Ein Token läuft 60 Minuten nach der Lösung ab.
`error`	—	Wird bei einem Netzwerk- oder Serverfehler ausgelöst.

Beispiel für die Auswertung:

```
<eu-captcha
  sitekey="EUCAPTCHA_SITE_KEY"
  (completed)="onToken($event)"
  (expired)="onExpired()"
  (error)="onError()"
/>
```

```
onToken(token: string): void {
  console.log('token:', token);
}

onExpired(): void {
  console.log('token expired');
}

onError(): void {
  console.log('challenge error');
}
```

Beispiele für die Darstellung:

```
<eu-captcha sitekey="EUCAPTCHA_SITE_KEY" theme="dark" />
<eu-captcha sitekey="EUCAPTCHA_SITE_KEY" [width]="280" />
<eu-captcha sitekey="EUCAPTCHA_SITE_KEY" [height]="60" />
```

5.3.4.6 Challenge verzögert starten

Setzen Sie `autostart` auf `false` und vergeben Sie eine Kennung:

```
<eu-captcha
  sitekey="EUCAPTCHA_SITE_KEY"
  widgetId="my-captcha"
  [autostart]="false"
/>

<button (click)="euCaptcha.execute('my-captcha')">Verify</button>
```

5.3.4.7 Zustand abfragen

Prüfen Sie vor dem Absenden, ob die Challenge bestanden ist:

```
import { isEuCaptchaDone } from '@myrasec/eu-captcha-angular21';

function handleSubmit(): void {
    if (!isEuCaptchaDone()) {
        // challenge not yet complete
        return;
    }
    // proceed with form submission
}
```

Alternativ warten Sie auf die Nachricht `euCaptchaDone` am Fenster:

```
import { Component, OnInit, OnDestroy } from '@angular/core';

@Component({ /* ... */ })
export class MyComponent implements OnInit, OnDestroy {
  private listener = (msg: MessageEvent) => {
    if (msg.data.type === 'euCaptchaDone') {
      // enable submit button, update state, etc.
    }
  };

  ngOnInit(): void {
    window.addEventListener('message', this.listener, false);
  }

  ngOnDestroy(): void {
    window.removeEventListener('message', this.listener, false);
  }
}
```

Die Prüfung im Browser steuert nur die Bedienung. Prüfen Sie das Token in jedem Fall zusätzlich auf dem Server. Siehe **Endpunkt /verify**.

5.3.4.8 Vollständiges Beispiel

Siehe **Angular und Java**.

5.3.5 Svelte

Das Paket `@myrasec/eu-captcha-svelte` stellt die Komponente `EuCaptcha` für Svelte 5 bereit.

5.3.5.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Svelte	Version 5, auch als Grundlage von SvelteKit
Sitekey	Öffentlicher Sitekey aus der Ansicht **Details**
Serverseite	Endpunkt, der das Token prüft

5.3.5.2 Paket installieren

Installieren Sie das Paket:

```
npm i @myrasec/eu-captcha-svelte
```

5.3.5.3 Komponente einbinden

Binden Sie die Komponente in das zu schützende Formular ein:

```
<script lang="ts">
  import { EuCaptcha, isEuCaptchaDone } from "@myrasec/eu-captcha-svelte";

  const captchaSitekey = "EUCAPTCHA_SITE_KEY";

  function handleSubmit(e: SubmitEvent): void {
    e.preventDefault();
    if (!isEuCaptchaDone()) {
      // challenge not yet complete
      return;
    }
    // proceed with form submission
  }
</script>

<form onsubmit={handleSubmit}>
  <!-- your fields -->
```

```
  <EuCaptcha sitekey={captchaSitekey} />
  <button type="submit">Submit</button>
</form>
```

Sie prüfen die Einbindung mit einem beliebigen erfundenen Sitekey. Zu einem unbekannten Sitekey läuft das CAPTCHA mit den Vorgabewerten, und der gesamte Verkehr wird durchgelassen.

5.3.5.4 Eigenschaften

Die folgenden Eigenschaften stehen zur Verfügung:

Eigenschaft	Typ	Vorgabe	Wirkung
`sitekey`	`string`	—	Öffentlicher Sitekey. Der Wert ist erforderlich.
`theme`	`string`	`"light"`	Darstellung, Werte `"light"` und `"dark"`.
`width`	`number`	`330`	Breite des Widgets in Pixeln.
`height`	`number`	`100`	Höhe des Widgets in Pixeln.
`widgetId`	`string`	—	Eigene Kennung des Widgets. Ohne Angabe erzeugt das Paket eine Kennung. Für `euCaptcha.execute()` ist der Wert erforderlich.
`autostart`	`boolean`	`true`	Startet die Challenge beim Einhängen der Komponente. Der Wert `false` verschiebt den Start auf den Aufruf `euCaptcha.execute(widgetId)`.
`onComplete`	`(token: string) => void`	—	Wird mit dem Token aufgerufen, sobald die Challenge bestanden ist.
		—	

Eigenschaft	Typ	Vorgabe	Wirkung
`onExpired`	`() => void`		Wird aufgerufen, wenn das Token abläuft. Ein Token läuft 60 Minuten nach der Lösung ab.
`onError`	`() => void`	—	Wird bei einem Netzwerk- oder Serverfehler aufgerufen.

Beispiele für die Darstellung:

```
<EuCaptcha sitekey={captchaSitekey} theme="dark" />
<EuCaptcha sitekey={captchaSitekey} width={280} />
<EuCaptcha sitekey={captchaSitekey} height={60} />
```

5.3.5.5 Rückrufe verwenden

```
<EuCaptcha
  sitekey={captchaSitekey}
  onComplete={(token) => console.log("token:", token)}
  onExpired={() => console.log("token expired")}
  onError={() => console.log("challenge error")}
/>
```

5.3.5.6 Challenge verzögert starten

Setzen Sie `autostart` auf `false` und vergeben Sie eine Kennung:

```
<EuCaptcha
  sitekey={captchaSitekey}
  widgetId="my-captcha"
  autostart={false}
/>

<button onclick={() => (window as any).euCaptcha.execute("my-captcha")}
>Verify</button>
```

5.3.5.7 Zustand abfragen

Prüfen Sie vor dem Absenden, ob die Challenge bestanden ist:

```
import { isEuCaptchaDone } from "@myrasec/eu-captcha-svelte";

function handleSubmit(e: SubmitEvent): void {
    e.preventDefault();
    if (!isEuCaptchaDone()) {
        // challenge not yet complete
        return;
    }
    // proceed with form submission
}
```

Alternativ warten Sie auf die Nachricht `euCaptchaDone` am Fenster:

```
import { onMount, onDestroy } from "svelte";

function listenForCaptchaDone(msg: MessageEvent): void {
    if (msg.data.type === "euCaptchaDone") {
        // enable submit button, update reactive state, etc.
    }
}

onMount(() => window.addEventListener("message", listenForCaptchaDone,
false));
onDestroy(() => window.removeEventListener("message", listenForCaptchaDone,
false));
```

5.3.5.8 SvelteKit

Die Komponente greift auf Schnittstellen des Browsers zu und richtet sich vollständig in `onMount` ein. Sie lässt sich daher ohne weitere Einstellungen auch in serverseitig gezeichneten Seiten einbinden. Eine Prüfung auf `browser` oder ein dynamischer Import ist nicht erforderlich.

Die Prüfung im Browser steuert nur die Bedienung. Prüfen Sie das Token in jedem Fall zusätzlich auf dem Server. Siehe **Endpunkt /verify**.

5.3.5.9 Vollständiges Beispiel

Siehe **Svelte und Ruby**.

5.4 Backend

5.4.1 PHP

Für PHP stehen zwei Pakete bereit. Beide prüfen das Token auf dem Server und geben dasselbe Ergebnisobjekt zurück.

5.4.1.1 Paket wählen

Wählen Sie das Paket zur PHP-Version Ihrer Anwendung:

Paket	PHP	Technik
`myra-security-gmbh/eu-captcha`	ab Version 8.0	Benannte Argumente, Typangaben, HTTP über Guzzle (`guzzlehttp/guzzle` ^6.0 oder ^7.0)
`myra-security-gmbh/eu-captcha-old`	ab Version 5.0	Ohne weitere Abhängigkeiten, HTTP über `file_get_contents()` mit einem Stream-Kontext

5.4.1.2 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
PHP	ab Version 8.0 für `eu-captcha` , ab Version 5.0 für `eu-captcha-old`
Composer	Zugriff auf die Kommandozeile des Servers
Zugangsdaten	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

5.4.1.3 Paket installieren

Installieren Sie das Paket für PHP 8:

```
composer require myra-security-gmbh/eu-captcha
```

Für PHP 5 bis PHP 7 installieren Sie stattdessen:

```
composer require myra-security-gmbh/eu-captcha-old
```

5.4.1.4 Widget einbinden

Ergänzen Sie jede Seite mit einem zu schützenden Formular um den Skript-Verweis:

```
<script src="https://cdn.eu-captcha.eu/verify.js" async defer></script>
```

Fügen Sie das Widget in das Formular ein:

```
<div class="eu-captcha" data-sitekey="EUCAPTCHA_SITE_KEY"></div>
```

Bei einer Single-Page-Anwendung mit React, Vue oder Angular binden Sie das Widget über das jeweilige npm-Paket ein. Das PHP-Paket übernimmt dann allein die serverseitige Prüfung.

5.4.1.5 Token prüfen

Prüfen Sie das übermittelte Token auf dem Server:

```
<?php

use Myrasec\EuCaptcha;

$captcha = new EuCaptcha(
    sitekey: EUCAPTCHA_SITE_KEY,
    secret:  EUCAPTCHA_SECRET_KEY,
);
```

```
$result = $captcha->validate();

if (!$result->success()) {
    // Reject the form submission
}
```

Mit `eu-captcha-old` übergeben Sie die Werte stattdessen als assoziatives Feld:

```
<?php

use Myrasec\EuCaptcha;

$captcha = new EuCaptcha([
    'sitekey' => EUCAPTCHA_SITE_KEY,
    'secret'  => EUCAPTCHA_SECRET_KEY,
]);

$result = $captcha->validate();
```

`validate()` liest das Token aus `$_POST['eu-captcha-response']`. Ist `$_POST` leer, wertet die Methode den Rumpf der Anfrage als JSON aus. Die IP-Adresse des Besuchers ermittelt die Methode aus den Kopfzeilen der Anfrage.

5.4.1.6 Optionen

Die folgenden Optionen stehen zur Verfügung:

Option	Typ	Vorgabe	Wirkung
`sitekey`	string	—	Öffentlicher Sitekey. Der Wert ist erforderlich.
`secret`	string	—	Geheimer Schlüssel. Der Wert ist erforderlich und darf nicht im Browser erscheinen.

Option	Typ	Vorgabe	Wirkung
`failDefault`	`bool`	`true`	Rückgabewert für den Netzwerk- und den Token-Zustand, wenn die API nicht erreichbar ist. `true` lässt die Absendung zu, `false` weist sie ab.
`checkCdnHeaders`	`bool`	`true`	Ermittelt die IP-Adresse des Besuchers aus den Kopfzeilen `HTTP_CLIENT_IP`, `HTTP_X_FORWARDED_FOR` und `HTTP_X_REAL_IP`, bevor `REMOTE_ADDR` verwendet wird. Setzen Sie den Wert auf `false`, wenn der Server nicht hinter einem vorgelagerten System steht oder Sie die IP-Adresse selbst übergeben.

Nur `eu-captcha` für PHP 8 kennt zusätzlich die folgenden Optionen:

Option	Typ	Vorgabe	Wirkung
`verifyUrl`	`string`	Adresse der Produktivumgebung	Überschreibt die Adresse des Endpunkts `/verify`. Verwenden Sie die Option für Tests.
`credentialsUrl`	`string`	Adresse der Produktivumgebung	Überschreibt die Adresse des Endpunkts `/verify-credentials`.
`client`	`?Client`	`null`	Eigene Instanz von Guzzle für abweichende Einstellungen oder für Tests.

5.4.1.7 Das Ergebnisobjekt

`validate()` gibt ein Objekt `EuCaptchaResult` mit drei Methoden zurück:

Methode	Gibt `true` zurück, wenn
`success()`	die API erreichbar war und das Token gültig ist.
`successNetwork()`	der Aufruf der API ohne Netzwerk- oder Übertragungsfehler abgeschlossen wurde.
	die API das übermittelte Token als gültig gemeldet hat.

Methode	Gibt `true` zurück, wenn
`successToken()`	

Die getrennte Abfrage unterscheidet eine nicht bestandene Challenge von einer Störung der API:

```php
<?php

$result = $captcha->validate();

if (!$result->successNetwork()) {
    // Could not reach the API — consider logging or alerting
}

if (!$result->successToken()) {
    // Token was rejected — the submission is likely automated
}
```

5.4.1.8 Token und IP-Adresse selbst übergeben

Bei abweichenden Feldnamen übergeben Sie Token und IP-Adresse selbst:

```php
<?php

$token    = $_POST['my-captcha-field'] ?? '';
$clientIp = $_SERVER['REMOTE_ADDR'];

$result = $captcha->validate($token, $clientIp);
```

5.4.1.9 Zugangsdaten prüfen

Mit `verifyCredentials()` prüfen Sie Sitekey und Secret ohne ein Token des Browsers, zum Beispiel beim Start der Anwendung:

```php
<?php

$captcha = new EuCaptcha(sitekey: EUCAPTCHA_SITE_KEY, secret:
EUCAPTCHA_SECRET_KEY);

if (!$captcha->verifyCredentials()) {
```

```
    // Credentials are invalid or the API is unreachable — log and alert
}
```

Bei einem Netzwerk- oder API-Fehler gibt die Methode `false` zurück und löst keine Ausnahme aus. Der Aufruf ist damit auch während der Initialisierung sicher.

Siehe **Endpunkt /verify-credentials**.

5.4.1.10 Symfony und Laravel

Für beide Rahmenwerke siehe **Symfony und Laravel**.

5.4.1.11 Vollständiges Beispiel

Siehe **React und PHP**.

5.4.2 Symfony und Laravel

Symfony und Laravel verwenden dasselbe Paket `myra-security-gmbh/eu-captcha` wie jede andere Anwendung mit PHP 8. Beide Rahmenwerke bringen eine eigene Ermittlung der Besucher-IP-Adresse mit, die Sie an `validate()` übergeben.

5.4.2.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
PHP	ab Version 8.0
Paket	`myra-security-gmbh/eu-captcha`
Zugangsdaten	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

Die Installation des Pakets ist unter **PHP** beschrieben.

5.4.2.2 Symfony

Verwenden Sie in Ihren Diensten und Controllern den Typ `EuCaptchaInterface`. Symfony verbindet den Dienst dann selbstständig, ohne dass Ihr Quelltext von der konkreten Klasse abhängt.

Tragen Sie den Dienst in `config/services.yaml` ein und verweisen Sie die Schnittstelle auf ihn:

```
services:
    Myrasec\EuCaptcha:
        arguments:
            $sitekey: '%env(EUCAPTCHA_SITE_KEY)%'
            $secret:  '%env(EUCAPTCHA_SECRET_KEY)%'
    Myrasec\EuCaptchaInterface: '@Myrasec\EuCaptcha'
```

Übergeben Sie die Schnittstelle an den Controller:

```
<?php

namespace App\Controller;

use Myrasec\EuCaptchaInterface;
```

```
use Symfony\Component\HttpFoundation\Request;
use Symfony\Component\HttpFoundation\Response;
use Symfony\Component\Routing\Attribute\Route;

class ContactController
{
    public function __construct(private EuCaptchaInterface $captcha) {}

    #[Route('/contact', methods: ['POST'])]
    public function submit(Request $request): Response
    {
        $result = $this->captcha->validate(
            $request->request->getString('eu-captcha-response'),
            $request->getClientIp() ?? '',
        );

        if (!$result->success()) {
            return new Response('CAPTCHA verification failed',
Response::HTTP_BAD_REQUEST);
        }

        // process the form...

        return new Response('OK');
    }
}
```

`$request->getClientIp()` beachtet die Einstellung der vertrauenswürdigen vorgelagerten Systeme von Symfony. Hinter einem CDN oder einem Lastverteiler erreicht damit die tatsächliche IP-Adresse des Besuchers die API. Als drittes Argument nimmt `validate()` zusätzlich die Kennung des Browsers entgegen.

5.4.2.3 Laravel

Legen Sie die Zugangsdaten in der Datei `.env` ab und veröffentlichen Sie sie über `config/services.php`.

Datei `.env`:

```
EUCAPTCHA_SITE_KEY=YOUR_SITEKEY
EUCAPTCHA_SECRET_KEY=YOUR_SECRET
```

Datei `config/services.php`:

```
'eucaptcha' => [
    'sitekey' => env('EUCAPTCHA_SITE_KEY'),
```

```
    'secret'  => env('EUCAPTCHA_SECRET_KEY'),
],
```

Prüfen Sie das Token im Controller:

```
<?php

namespace App\Http\Controllers;

use Myrasec\EuCaptcha;
use Illuminate\Http\Request;
use Illuminate\Http\RedirectResponse;

class ContactController extends Controller
{
    public function submit(Request $request): RedirectResponse
    {
        $captcha = new EuCaptcha(
            sitekey: config('services.eucaptcha.sitekey'),
            secret:  config('services.eucaptcha.secret'),
        );

        $result = $captcha->validate(
            $request->input('eu-captcha-response'),
            $request->ip(),
        );

        if (!$result->success()) {
            return back()->withErrors(['captcha' => 'CAPTCHA verification
failed.']);
        }

        // process the form...

        return redirect()->route('contact.success');
    }
}
```

`$request->ip()` beachtet die Einstellung der vertrauenswürdigen vorgelagerten Systeme von Laravel.

5.4.2.4 Prüfung in einem Form Request

Für mehrere Controller legen Sie die Prüfung in einen eigenen `FormRequest`:

```
<?php

namespace App\Http\Requests;
```

```
use Myrasec\EuCaptcha;
use Illuminate\Foundation\Http\FormRequest;
use Illuminate\Contracts\Validation\Validator;

class ContactRequest extends FormRequest
{
    public function rules(): array
    {
        return [
            'name'    => ['required', 'string', 'max:255'],
            'email'   => ['required', 'email'],
            'message' => ['required', 'string'],
        ];
    }

    protected function withValidator(Validator $validator): void
    {
        $validator->after(function (Validator $validator) {
            $captcha = new EuCaptcha(
                sitekey: config('services.eucaptcha.sitekey'),
                secret:  config('services.eucaptcha.secret'),
            );

            $result = $captcha->validate(
                $this->input('eu-captcha-response'),
                $this->ip(),
            );

            if (!$result->success()) {
                $validator->errors()->add('captcha', 'CAPTCHA verification
failed.');
            }
        });
    }
}
```

Übergeben Sie der Methode des Controllers anschließend `ContactRequest` statt `Request`. Laravel prüft die Anfrage dann vor dem Aufruf der Methode:

```
public function submit(ContactRequest $request): RedirectResponse
{
    // validation and CAPTCHA check already passed
    // process the form...

    return redirect()->route('contact.success');
}
```

5.4.3 Python

Das Paket `myra-eucaptcha` prüft das Token auf dem Server. Es stellt eine synchrone und eine asynchrone Schnittstelle bereit.

5.4.3.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Paketverwaltung	`pip` oder `uv`
Zugangsdaten	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

5.4.3.2 Paket installieren

Installieren Sie das Paket aus PyPI:

```
pip install myra-eucaptcha
```

Mit der Paketverwaltung `uv` lautet der Befehl:

```
uv add myra-eucaptcha
```

5.4.3.3 Widget einbinden

Fügen Sie das Widget in das zu schützende Formular ein:

```
<div class="eu-captcha" data-sitekey="EUCAPTCHA_SITE_KEY"></div>
```

Ergänzen Sie die Seite um den Skript-Verweis:

```
<script src="https://cdn.eu-captcha.eu/verify.js" async defer></script>
```

Das Skript zeichnet das Widget, führt die Challenge aus und ergänzt das Formular um das Token.

Eine vollständige Seite sieht damit wie folgt aus:

```
<html>
    <head>
        <title>captcha integration sample</title>
    </head>
    <body>

    <form action="/login" method="post">
        <table>
            <tr>
                <td>user</td>
                <td>
                    <input type=text name="username" />
                </td>
            </tr>
            <tr>
                <td>pass</td>
                <td>
                    <input type=password name="password" />
                </td>
            </tr>
            <tr>
                <td colspan="2">
                    <div class="eu-captcha" data-
sitekey="EUCAPTCHA_SITE_KEY"></div>
                </td>
            </tr>
            <tr><td colspan="2"><input type="submit" name="submit" /></td></
tr>
        </table>
    </form>

    <script src="https://cdn.eu-captcha.eu/verify.js" async defer></script>
    </body>
</html>
```

5.4.3.4 Token prüfen

Das folgende Beispiel zeigt die serverseitige Prüfung mit FastAPI:

```
from myra_eucaptcha import MyraEuCaptchaClient, MyraEuCaptchaClientConfig

@app.post("/login", response_class=HTMLResponse)
async def login(
    request:Request,
```

```
    username: str = Form(...),
    password: str = Form(...),
    eu_captcha_response: str = Form(..., alias="eu-captcha-response"),
):

    # configure the captcha settings and behaviour
    captcha_config = MyraEuCaptchaClientConfig(
        sitekey="EUCAPTCHA_SITE_KEY",
        secret="EUCAPTCHA_SECRET_KEY",
    )

    client = MyraEuCaptchaClient(config=captcha_config)

    # this is depending on your webserver-setup and whether we're behind
    # a reverse-proxy (nginx, apache, traefik, or something similar.)
    remote_addr = request.headers.get('x-real-ip','')
    validationresult = await client.avalidate(
                token=eu_captcha_response,
                remote_addr=remote_addr)

    if validationresult.success:
        # yay - all good.
        # <your-code-here>
        pass
    else:
        # validation failed, check validationresult.errors
        # <your-code-here>
        pass
```

Die asynchrone Prüfung erfolgt über `avalidate()`, die synchrone über `validate()`.

Woher die IP-Adresse des Besuchers stammt, hängt von Ihrem Webserver ab. Steht die Anwendung hinter einem vorgelagerten System wie nginx, Apache oder Traefik, lesen Sie die Adresse aus der entsprechenden Kopfzeile.

5.4.3.5 Verhalten bei Fehlern

In der Voreinstellung ist das Paket fehlertolerant: Bei einer Zeitüberschreitung, einem API-Fehler oder einem Netzwerkfehler liefert die Prüfung das Ergebnis `True`, und es wird keine Ausnahme ausgelöst. Die aufgetretenen Fehler stehen in `MyraEuCaptchaResult.errors`.

Sie ändern dieses Verhalten über `MyraEuCaptchaClientConfig`:

```
    captcha_config = MyraEuCaptchaClientConfig(
        sitekey="EUCAPTCHA_SITE_KEY",
        secret="EUCAPTCHA_SECRET_KEY",
        connect_timeout: int = 3
        read_timeout: int = 10
```

```
        write_timeout: int = 10
        pool_timeout: int = 3
        default_result_on_error:bool = True ## this is the validation result,
that is returned on exceptions
        suppress_exceptions:bool = True     ## if you want exceptions to be
raised, set this to False
    )
```

Einstellung	Vorgabe	Wirkung
`connect_timeout`	`3`	Zeitgrenze für den Verbindungsaufbau in Sekunden.
`read_timeout`	`10`	Zeitgrenze für das Lesen der Antwort in Sekunden.
`write_timeout`	`10`	Zeitgrenze für das Senden der Anfrage in Sekunden.
`pool_timeout`	`3`	Zeitgrenze für eine freie Verbindung aus dem Verbindungsspeicher in Sekunden.
`default_result_on_error`	`True`	Ergebnis der Prüfung, wenn eine Ausnahme auftritt.
`suppress_exceptions`	`True`	Auf `False` löst das Paket die Ausnahme aus, statt sie zu unterdrücken.

Für Django siehe **Django**.

5.4.3.6 Vollständiges Beispiel

Siehe **Vue und Python**.

5.4.4 Django

Das Paket `myra-eucaptcha-django` stellt ein Formularfeld, ein Widget und eine Verwaltung der Konfigurationen im Administrationsbereich von Django bereit.

5.4.4.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Paketverwaltung	`pip` oder `uv`
Zugang	Administrationsbereich von Django
Zugangsdaten	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

5.4.4.2 Paket installieren

Um das Paket einzurichten, gehen Sie wie folgt vor:

- Installieren Sie das Paket:

```
pip install myra-eucaptcha-django
```

Mit der Paketverwaltung `uv` lautet der Befehl:

```
uv add myra-eucaptcha-django
```

- Tragen Sie die Anwendung in `settings.py` ein:

```
# settings.py
INSTALLED_APPS = [
    ...
    'myra_eucaptcha_django',
]
```

- Führen Sie die Migrationen aus:

```
python manage.py migrate
```

→ Der Administrationsbereich zeigt den Eintrag **Captcha Configurations** an.

5.4.4.3 Konfiguration anlegen

Um eine Konfiguration anzulegen, gehen Sie wie folgt vor:

- Öffnen Sie im Administrationsbereich den Eintrag **Captcha Configurations**.
- Legen Sie eine Konfiguration mit Ihrem Sitekey und Ihrem Secret an.
- Markieren Sie die Konfiguration als **Default**, damit Formulare sie ohne Angabe eines Namens verwenden.

→ Die Konfiguration steht in den Formularen zur Verfügung.

5.4.4.4 Feld in ein Formular einbauen

Ergänzen Sie das Formular um das Feld `CaptchaField` und um `CaptchaFormMixin`:

```
from django import forms
from myra_eucaptcha_django import CaptchaField, CaptchaFormMixin

class ContactForm(CaptchaFormMixin, forms.Form):
    name = forms.CharField()
    email = forms.EmailField()
    message = forms.CharField(widget=forms.Textarea)

    # Uses the default configuration
    captcha = CaptchaField()
```

Übergeben Sie in der Ansicht die Anfrage, damit das Feld die IP-Adresse des Besuchers prüfen kann:

```
from django.shortcuts import render, redirect
from .forms import ContactForm

def contact_view(request):
    if request.method == "POST":
        # Pass request= to enable IP validation
        form = ContactForm(request.POST, request=request)
        if form.is_valid():
```

```
            # Process the form
            return redirect("success")
    else:
        form = ContactForm()

    return render(request, "contact.html", {"form": form})
```

Zeichnen Sie das Formular in der Vorlage:

```
<form method="post">
    {% csrf_token %}
    {{ form.as_p }}
    <button type="submit">Send</button>
</form>
```

Das Widget bringt das erforderliche JavaScript selbst mit.

5.4.4.5 Mehrere Konfigurationen verwenden

Sie legen im Administrationsbereich mehrere Konfigurationen mit eindeutigen Namen an, zum Beispiel `default`, `contact-form` und `registration`. Im Formular geben Sie den Namen der gewünschten Konfiguration an:

```
class ContactForm(CaptchaFormMixin, forms.Form):
    name = forms.CharField()
    captcha = CaptchaField("contact-form")

class RegistrationForm(CaptchaFormMixin, forms.Form):
    username = forms.CharField()
    captcha = CaptchaField("registration")

class CommentForm(CaptchaFormMixin, forms.Form):
    comment = forms.CharField()
    captcha = CaptchaField()  # Uses default configuration
```

5.4.4.6 Felder einer Konfiguration

Die folgenden Felder stehen zur Verfügung:

Feld	Vorgabe	Wirkung
`name`	erforderlich	Eindeutiger Name der Konfiguration.

Feld	Vorgabe	Wirkung
`description`	—	Freie Notiz zur Verwendung.
`sitekey`	erforderlich	Öffentlicher Sitekey für das Widget.
`secret`	erforderlich	Geheimer Schlüssel für die serverseitige Prüfung.
`is_default`	`False`	Verwendet die Konfiguration, wenn das Formular keinen Namen angibt.
`is_active`	`True`	Gibt die Konfiguration zur Verwendung frei.

Die folgenden Felder ändern die Adressen der Dienste:

Feld	Vorgabe	Wirkung
`verify_url`	`https://api.eu-captcha.eu/v1/verify/`	Adresse des Endpunkts für die Prüfung.
`widget_url`	`https://cdn.eu-captcha.eu/verify.js`	Adresse des Skripts für das Widget.

Die folgenden Felder setzen die Zeitgrenzen in Sekunden:

Feld	Vorgabe	Wirkung
`connect_timeout`	`3`	Zeitgrenze für den Verbindungsaufbau.
`read_timeout`	`10`	Zeitgrenze für das Lesen der Antwort.
`write_timeout`	`10`	Zeitgrenze für das Senden der Anfrage.
`pool_timeout`	`3`	Zeitgrenze für eine freie Verbindung aus dem Verbindungsspeicher.

Die folgenden Felder steuern das Verhalten bei Fehlern:

Feld	Vorgabe	Wirkung
`default_result_on_error`	`True`	Gibt bei einem Netzwerkfehler ein positives Ergebnis zurück und lässt die Absendung zu.
`suppress_exceptions`	`True`	Unterdrückt Ausnahmen und gibt stattdessen das Ergebnis zurück.

5.4.4.7 Konfiguration über die Einstellungen

Statt der Datenbank verwenden Sie die Datei `settings.py` :

```
# settings.py
EUCAPTCHA_SITEKEY = "EUCAPTCHA_SITE_KEY"
EUCAPTCHA_SECRET = "EUCAPTCHA_SECRET_KEY"

# Optional
EUCAPTCHA_VERIFY_URL = "https://api.eu-captcha.eu/v1/verify/"
EUCAPTCHA_WIDGET_URL = "https://cdn.eu-captcha.eu/verify.js"
EUCAPTCHA_CONNECT_TIMEOUT = 3
EUCAPTCHA_READ_TIMEOUT = 10
EUCAPTCHA_WRITE_TIMEOUT = 10
EUCAPTCHA_POOL_TIMEOUT = 3
EUCAPTCHA_DEFAULT_RESULT_ON_ERROR = True
EUCAPTCHA_SUPPRESS_EXCEPTIONS = True
```

Konfigurationen aus der Datenbank haben Vorrang vor den Einstellungen.

5.4.4.8 Schnittstelle

Baustein	Verwendung
`CaptchaField(config_name=None, **kwargs)`	Formularfeld, das das Widget zeichnet und die Antwort prüft. Ohne `config_name` gilt die Standardkonfiguration.
`CaptchaFormMixin`	Reicht die Anfrage an die Felder weiter, damit die IP-Adresse geprüft werden kann.
`CaptchaWidget`	Zeichnet die Challenge. Der Baustein wird in der Regel nicht selbst verwendet.

Baustein	Verwendung
`validate_captcha(...)`	Prüft ein Token unmittelbar. Bei einem Fehler löst die Funktion `django.core.exceptions.ValidationError` aus.

```python
from myra_eucaptcha_django import validate_captcha

validate_captcha(
    token="captcha-response-token",
    remote_addr="client-ip",  # Optional
    config_name="my-config",  # Optional
)
```

5.4.4.9 Vollständiges Beispiel

Siehe **HTML und Django**.

5.4.5 Ruby

Das Gem `eu_captcha` prüft das Token auf dem Server. Es kommt ohne weitere Abhängigkeiten aus und verwendet `net/http` aus der Standardbibliothek von Ruby.

5.4.5.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Ruby	ab Version 2.7
Zugangsdaten	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

5.4.5.2 Gem installieren

Installieren Sie das Gem:

```
gem install eu_captcha
```

Alternativ ergänzen Sie die Datei `Gemfile`:

```
gem 'eu_captcha'
```

5.4.5.3 Widget einbinden

Ergänzen Sie jede Seite mit einem zu schützenden Formular um den Skript-Verweis:

```
<script src="https://cdn.eu-captcha.eu/verify.js" async defer></script>
```

Fügen Sie das Widget in das Formular ein:

```
<div class="eu-captcha" data-sitekey="EUCAPTCHA_SITE_KEY"></div>
```

Bei einer Single-Page-Anwendung mit React, Vue oder Angular binden Sie das Widget über das jeweilige npm-Paket ein. Das Gem übernimmt dann allein die serverseitige Prüfung.

5.4.5.4 Token prüfen

Prüfen Sie das übermittelte Token auf dem Server:

```ruby
require 'eu_captcha'

captcha = EuCaptcha::Client.new(
  sitekey: ENV['EUCAPTCHA_SITE_KEY'],
  secret:  ENV['EUCAPTCHA_SECRET_KEY']
)

result = captcha.validate(
  token:       params['eu-captcha-response'],
  remote_addr: request.ip
)

render_error unless result.success?
```

5.4.5.5 Optionen

Die folgenden Optionen übergeben Sie als Schlüsselwortargumente an den Konstruktor:

Option	Typ	Vorgabe	Wirkung
`sitekey`	String	—	Öffentlicher Sitekey. Der Wert ist erforderlich.
`secret`	String	—	Geheimer Schlüssel. Der Wert ist erforderlich und darf nicht im Browser erscheinen.
`fail_default`	Boolean	`true`	Rückgabewert für den Netzwerk- und den Token-Zustand, wenn die API nicht erreichbar ist. `true` lässt die Absendung zu, `false` weist sie ab.
`check_cdn_`	Boolean	`true`	Ermittelt die IP-Adresse des Besuchers aus den Kopfzeilen `HTTP_CLIENT_IP` , `HTTP_X_FORWARDED_FOR`

Option	Typ	Vorgabe	Wirkung
`headers`			und `HTTP_X_REAL_IP` der Rack-Umgebung, bevor `REMOTE_ADDR` verwendet wird.
`verify_url`	String	Adresse der Produktivumgebung	Überschreibt die Adresse des Endpunkts `/verify`. Verwenden Sie die Option für Tests.
`credentials_url`	String	Adresse der Produktivumgebung	Überschreibt die Adresse des Endpunkts `/verify-credentials`.

5.4.5.6 Das Ergebnisobjekt

`validate` gibt ein Objekt `EuCaptcha::Result` mit den folgenden Methoden zurück:

Methode	Rückgabe
`success?`	`true`, wenn die API erreichbar war und das Token gültig ist.
`success_network?`	`true`, wenn der Aufruf der API ohne Netzwerk- oder Übertragungsfehler abgeschlossen wurde.
`success_token?`	`true`, wenn die API das übermittelte Token als gültig gemeldet hat.
`train`	`true`, `false` oder `nil`.

Die getrennte Abfrage unterscheidet eine nicht bestandene Challenge von einer Störung der API:

```
result = captcha.validate(token: params['eu-captcha-response'], remote_addr:
request.ip)

unless result.success_network?
  # Could not reach the API — consider logging or alerting
end
```

```
unless result.success_token?
  # Token was rejected – the submission is likely automated
end
```

`train` gibt `true` zurück, wenn die API die eigentliche Prüfung übersprungen und ein positives Ergebnis erzwungen hat. Das geschieht, wenn der Sitekey nicht besteht, das Secret nicht dazu passt oder der Schutz des Sitekeys abgeschaltet ist. `success?` wertet dieses Kennzeichen bereits aus und gibt in diesem Fall `false` zurück. Bei einem Netzwerkfehler gibt `train` den Wert `nil` zurück.

5.4.5.7 Zugangsdaten prüfen

Mit `verify_credentials` prüfen Sie Sitekey und Secret ohne ein Token des Browsers, zum Beispiel beim Start der Anwendung:

```
captcha = EuCaptcha::Client.new(
  sitekey: ENV['EUCAPTCHA_SITE_KEY'],
  secret:  ENV['EUCAPTCHA_SECRET_KEY']
)

unless captcha.verify_credentials
  # Credentials are invalid or the API is unreachable – log and alert
end
```

Bei einem Netzwerk- oder API-Fehler gibt die Methode `false` zurück und löst keine Ausnahme aus.

Siehe **Endpunkt /verify-credentials**.

5.4.5.8 Ruby on Rails

Legen Sie die Zugangsdaten in `config/credentials.yml.enc` ab und erzeugen Sie den Client in einer Initialisierungsdatei.

Datei `config/initializers/eu_captcha.rb`:

```
EU_CAPTCHA = EuCaptcha::Client.new(
  sitekey: Rails.application.credentials.dig(:eucaptcha, :sitekey),
```

```
  secret:  Rails.application.credentials.dig(:eucaptcha, :secret)
)
```

Prüfen Sie das Token im Controller:

```
class ContactController < ApplicationController
  def create
    result = EU_CAPTCHA.validate(
      token:       params['eu-captcha-response'],
      remote_addr: request.ip,
      user_agent:  request.user_agent
    )

    unless result.success?
      render json: { error: 'CAPTCHA verification failed' },
status: :unprocessable_entity
      return
    end

    # process the form...
  end
end
```

`request.ip` beachtet die Einstellung der vertrauenswürdigen vorgelagerten Systeme von Rails.

5.4.5.9 Sinatra und Rack

Übergeben Sie die Rack-Umgebung, damit das Gem die IP-Adresse und die Kennung des Browsers selbst ermittelt:

```
require 'sinatra'
require 'eu_captcha'

CAPTCHA = EuCaptcha::Client.new(
  sitekey: ENV['EUCAPTCHA_SITE_KEY'],
  secret:  ENV['EUCAPTCHA_SECRET_KEY']
)

post '/contact' do
  result = CAPTCHA.validate(
    token:    params['eu-captcha-response'],
    rack_env: env
  )

  halt 422, 'CAPTCHA verification failed' unless result.success?
```

```
  # process the form...
end
```

Bei einer übergebenen Rack-Umgebung beachtet `validate` die Einstellung `check_cdn_headers`.

5.4.5.10 Vollständiges Beispiel

Siehe **Svelte und Ruby**.

5.4.6 Java

Für Java stehen drei Clients bereit. Alle drei sind aus derselben OpenAPI-Beschreibung erzeugt und rufen denselben Endpunkt auf. Sie unterscheiden sich in der verwendeten Technik von Spring.

5.4.6.1 Client wählen

Anwendung	Client
Spring Boot 3 mit Jakarta EE	`java-webflux-boot3`
Spring Boot 2 oder Spring 5, reaktiv	`java-webflux-boot2`
Spring MVC auf dem Servlet-Stapel, Spring 6	`java-resttemplate`

Client	Voraussetzungen	Schnittstelle
`java-resttemplate`	Java 17, Spring Framework 6	Synchron: `VerifyResponse verifyClientToken(VerifyRequest)`. Wiederholt Anfragen bei den Statuscodes 5xx und 429 mit wachsendem Abstand.
`java-webflux-boot2`	Java 8, Spring Boot 2 mit Spring WebFlux	Reaktiv: `Mono<VerifyResponse> verifyClientToken(VerifyRequest)`
`java-webflux-boot3`	Java 17, Spring Boot 3 mit Jakarta EE 10	Reaktiv: `Mono<VerifyResponse> verifyClientToken(VerifyRequest)`. Verwendet durchgehend Anmerkungen aus `jakarta.*`.

5.4.6.2 Client einbinden

Binden Sie den gewählten Client mit Maven ein, hier am Beispiel von `java-resttemplate`:

```
<dependency>
  <groupId>com.myrasec</groupId>
  <artifactId>eu-captcha-java-resttemplate</artifactId>
```

```
  <version>1.0.0</version>
  <scope>compile</scope>
</dependency>
```

Mit Gradle lautet der Eintrag:

```
implementation "com.myrasec:eu-captcha-java-resttemplate:1.0.0"
```

Aus den Quellen erzeugen Sie den Client mit:

```
mvn clean install
```

5.4.6.3 Token prüfen

Das folgende Beispiel zeigt die synchrone Prüfung mit `java-resttemplate`:

```
import com.myrasec.client.ApiClient;
import com.myrasec.client.api.EuCaptchaApi;
import com.myrasec.client.model.VerifyRequest;
import com.myrasec.client.model.VerifyResponse;
import org.springframework.web.client.RestClientException;

ApiClient client = new ApiClient();
// client.setMaxAttemptsForRetry(3); // optional: retry up to 3 times on 5xx /
429

EuCaptchaApi api = new EuCaptchaApi(client);

VerifyRequest request = new VerifyRequest()
        .sitekey("YOUR_SITEKEY")
        .secret("YOUR_SECRET")
        .clientIp(clientIp)              // real end-user IP – see below
        .clientToken(euCaptchaToken)     // value of the "eu-captcha-response"
POST field
        .clientUserAgent(userAgent);     // value of the User-Agent request
header

try {
    VerifyResponse response = api.verifyClientToken(request);

    if (response.isTrainingMode()) {
        // Training mode: real validation was not performed – always allow.
        // Occurs when the sitekey does not exist, the secret is wrong,
        // or the sitekey is configured with train=true.
        allowAccess();
```

```
    } else if (Boolean.TRUE.equals(response.getSuccess())) {
        allowAccess();
    } else {
        denyAccess();
    }
} catch (RestClientException e) {
    // Network error or unexpected HTTP status.
    // Decide your fallback policy: allow or deny.
    log.error("EU CAPTCHA verification failed: {}", e.getMessage());
}
```

5.4.6.4 Reaktive Prüfung

Die beiden Clients für WebFlux geben ein `Mono<VerifyResponse>` zurück. Fügen Sie dieses in Ihre Verarbeitungskette ein, statt `.block()` aufzurufen:

```
import com.myrasec.client.ApiClient;
import com.myrasec.client.api.EuCaptchaApi;
import com.myrasec.client.model.VerifyRequest;
import com.myrasec.client.model.VerifyResponse;
import
org.springframework.web.reactive.function.client.WebClientResponseException;
import reactor.core.publisher.Mono;

ApiClient client = new ApiClient(); // thread-safe; share a single instance

EuCaptchaApi api = new EuCaptchaApi(client);

VerifyRequest request = new VerifyRequest()
        .sitekey("YOUR_SITEKEY")
        .secret("YOUR_SECRET")
        .clientIp(clientIp)
        .clientToken(euCaptchaToken)
        .clientUserAgent(userAgent);

Mono<VerifyResponse> result = api.verifyClientToken(request)
        .map(response -> {
            if (response.isTrainingMode()) {
                allowAccess();
            } else if (Boolean.TRUE.equals(response.getSuccess())) {
                allowAccess();
            } else {
                denyAccess();
            }
            return response;
        })
        .onErrorResume(WebClientResponseException.class, e -> {
            log.error("EU CAPTCHA verification failed: {}", e.getMessage());
            return Mono.empty();
        });
```

5.4.6.5 IP-Adresse des Besuchers ermitteln

Übergeben Sie stets die tatsächliche IP-Adresse des Besuchers, nicht die Adresse Ihres vorgelagerten Systems.

In einer Anwendung auf dem Servlet-Stapel:

```
// Without a proxy
String clientIp = httpServletRequest.getRemoteAddr();

// Behind a reverse proxy or CDN (use the header your provider documents)
String forwarded = httpServletRequest.getHeader("X-Forwarded-For");
if (forwarded != null && !forwarded.isBlank()) {
    // X-Forwarded-For is a comma-separated list; the leftmost entry is the
client IP
    clientIp = forwarded.split(",")[0].trim();
}
```

In einer Anwendung mit WebFlux:

```
// In a Spring WebFlux handler (ServerWebExchange)
String clientIp =
exchange.getRequest().getRemoteAddress().getAddress().getHostAddress();

// Behind a reverse proxy or CDN (use the header your provider documents)
String forwarded = exchange.getRequest().getHeaders().getFirst("X-Forwarded-
For");
if (forwarded != null && !forwarded.isBlank()) {
    clientIp = forwarded.split(",")[0].trim();
}
```

Die Kopfzeile `X-Forwarded-For` enthält eine durch Kommata getrennte Liste. Der erste Eintrag ist die IP-Adresse des Besuchers.

5.4.6.6 Felder der Anfrage

Feld	Typ	Pflicht	Inhalt
`sitekey`	String	ja	Öffentlicher Sitekey aus der Ansicht **Details**.
`secret`	String	ja	Geheimer Schlüssel zum Sitekey.

Feld	Typ	Pflicht	Inhalt
`clientIp`	String	ja	IPv4- oder IPv6-Adresse des Besuchers.
`clientToken`	String	ja	Token aus `verify.js`. Der Wert darf leer sein.
`clientUserAgent`	String	ja	Wert der Kopfzeile `User-Agent` aus der Anfrage des Besuchers.

5.4.6.7 Felder der Antwort

Feld	Typ	Inhalt
`success`	Boolean	`true`, wenn die Challenge bestanden wurde.
`train`	Boolean	`true`, wenn der Trainingsbetrieb aktiv war. Die Methode `isTrainingMode()` liest dieses Feld.

Der Endpunkt liegt unter `https://api.eu-captcha.eu/v1` und lautet `POST /verify`. Eine Anmeldung ist nicht erforderlich. Die Anfrage weist sich über das Feld `secret` aus.

5.4.6.8 Sicherheit

Schalten Sie `ApiClient.setDebugging(true)` nicht im Produktivbetrieb ein. Im Fehlersuchbetrieb schreibt der Client den vollständigen JSON-Rumpf in das Protokoll. `VerifyRequest.toString()` verbirgt das Secret, der tatsächlich übertragene Rumpf jedoch nicht.

Legen Sie das Secret in einer Umgebungsvariablen oder in einer Schlüsselverwaltung ab, niemals im Quelltext.

`ApiClient` ist nebenläufigkeitssicher. Erzeugen Sie eine Instanz und verwenden Sie sie in der gesamten Anwendung, statt je Anfrage eine neue Instanz anzulegen.

5.4.6.9 Vollständiges Beispiel

Siehe **Angular und Java**.

5.4.7 Client aus der Spezifikation

Für Sprachen ohne fertiges Paket erzeugen Sie den Client selbst aus der Spezifikation der Verifikations-API. Grundlage ist die Datei `openapi.yaml` nach OpenAPI 3.1.0. Sie beschreibt die Endpunkte `POST /verify` und `POST /verify-credentials` des Servers `https://api.eu-captcha.eu/v1`.

Für PHP, Symfony und Laravel, Python, Django, Ruby und Java stehen fertige Pakete zur Verfügung. Erzeugen Sie einen eigenen Client nur für Sprachen, die dort nicht aufgeführt sind. Siehe **Integration**.

5.4.7.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Spezifikation	Die Datei `openapi.yaml` liegt vor
Werkzeug	`openapi-generator-cli`; Myra erzeugt die eigenen Clients mit der Version 7.18.0

5.4.7.2 Client erzeugen

Um einen Client zu erzeugen, führen Sie den folgenden Befehl aus:

```
openapi-generator-cli generate -i openapi.yaml -g <generator> -o
<ausgabeverzeichnis>
```

Ersetzen Sie `<generator>` durch den Namen des gewünschten Generators. Die folgenden Beispiele zeigen die von Myra verwendeten Aufrufe:

Ziel	Befehl
Spring	`openapi-generator-cli generate -i openapi.yaml -g spring -o spring`
Node.js mit TypeScript	`openapi-generator-cli generate -i openapi.yaml -g typescript-node -o typescript-node`

Für weitere Sprachen — etwa Go, C# oder Rust — setzen Sie den Namen des jeweiligen Generators ein. Die Liste der Generatoren gehört zum Werkzeug, nicht zur Spezifikation.

5.4.7.3 Benennung angleichen

Damit die erzeugten Pakete zu den Paketen von Myra passen, übergeben Sie den Namensraum und den Paketnamen als zusätzliche Eigenschaften:

```
openapi-generator-cli generate -i openapi.yaml -g python \
  --additional-properties=invokerPackage=Myrasec\
\EuCaptcha,packageName=eucaptcha,library=httpx \
  -o python
```

Für Java verwendet Myra die folgenden Eigenschaften:

Variante	Zusätzliche Eigenschaften
`java-resttemplate`	`packageName=eucaptcha,groupId=com.myrasec,invokerPackage=com.myrasec.client,apiPackage=com.myrasec.client.api,modelPackage=com.myrasec.client.model,library=resttemplate`
`java-webclient`	`packageName=eucaptcha,groupId=com.myrasec,invokerPackage=com.myrasec.client,apiPackage=com.myrasec.client.api,modelPackage=com.myrasec.client.model,library=webclient`
`java-springboot3`	`packageName=eucaptcha,groupId=com.myrasec,invokerPackage=com.myrasec.client,apiPackage=com.myrasec.client.api,modelPackage=com.myrasec.client.model,library=webclient,useSpringBoot3=true,useJakartaEe=true`

Alle drei Varianten verwenden den Generator `java` . Siehe **Java**.

5.4.7.4 Erzeugten Client verwenden

Der Generator legt für den Endpunkt `POST /verify` die Operation `verifyClientToken` an. Die Namen in der Zielsprache richten sich nach den Regeln des jeweiligen Generators, der Aufbau der Anfrage und der Antwort bleibt jedoch gleich:

Feld der Anfrage	Beschreibung
`sitekey`	Öffentlicher Sitekey aus der Ansicht **Details**.

Feld der Anfrage	Beschreibung
`secret`	Zum Sitekey gehörendes Secret.
`client_ip`	Tatsächliche IP-Adresse des Besuchers.
`client_token`	Token aus `verify.js`.
`client_user_agent`	Kopfzeile `User-Agent` der Anfrage.

Siehe **Client-Token prüfen**.

Werten Sie in der Antwort stets auch das Feld `train` aus. Hat es den Wert `true`, hat keine Prüfung stattgefunden. Siehe **Sitekey konfigurieren**.

5.4.7.5 Widget einbinden

Der erzeugte Client übernimmt die serverseitige Prüfung. Das Widget in der Seite mit dem zu schützenden Formular ergänzen Sie unverändert. Siehe **HTML und JavaScript**.

5.5 Mobile Apps

5.5.1 Android

Das SDK für Android bettet das Widget in Ihre App ein und meldet den Zustand der Challenge über einen Empfänger.

5.5.1.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Android	ab Version 4.1 Jelly Bean, API-Ebene 16
Sprache	Kotlin oder Java. Das SDK ist in Kotlin geschrieben.
Zugangsdaten	Öffentlicher Sitekey aus der Ansicht **Details**

5.5.1.2 SDK einbinden

Ergänzen Sie die Datei `build.gradle` :

```
repositories {
    mavenCentral()
}

dependencies {
    implementation "eu.eucaptcha.android:eu-captcha-android:1.0.0"
    // Or for `build.gradle.kts`
    // implementation("eu.eucaptcha.android:eu-captcha-android:1.0.0")
}
```

5.5.1.3 Widget erzeugen

Erzeugen Sie das SDK und das Widget in Ihrer Activity:

```
import eu.eucaptcha.android.sdk.*

const val EU_CAPTCHA_SITEKEY = "EUCAPTCHA_SITE_KEY"
```

```
class MainActivity : ComponentActivity() {
    private val sdk by lazy {
        EuCaptchaSDK(context = this)
    }

    private val widget by lazy {
        sdk.createWidget(sitekey = EU_CAPTCHA_SITEKEY)
    }
}
```

5.5.1.4 Widget anzeigen

Zeigen Sie das Widget in Ihrer Oberfläche an und werten Sie den Zustand aus:

```
val captchaResponse = remember { mutableStateOf("") }
var buttonEnabled by remember { mutableStateOf(false) }

widget.setOnStateChangeListener { event ->
    captchaResponse.value = event.response
    when (event.state) {
        "completed" -> buttonEnabled = true
        "expired"   -> buttonEnabled = false
        "error"     -> buttonEnabled = true
    }
}

AndroidView(factory = { _ -> widget.view })

Button(onClick = { widget.reset() }, enabled = buttonEnabled) {
    Text("Submit")
}
```

Die folgenden Zustände meldet der Empfänger:

Zustand	Bedeutung
`completed`	Die Challenge ist bestanden. Das Feld `response` enthält das Token.
`expired`	Das Token ist abgelaufen.
`error`	Bei der Prüfung ist ein Fehler aufgetreten.

Der Aufruf `widget.reset()` startet eine neue Challenge.

5.5.1.5 Token prüfen

Senden Sie das Token an Ihren Server und prüfen Sie es dort. Siehe **Endpunkt /verify**.

5.5.2 iOS

Das SDK für iOS bettet das Widget in eine `WKWebView` ein und stellt eine Schnittstelle in Swift bereit.

5.5.2.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
iOS	ab Version 13.0
Swift	ab Version 5.5
Xcode	ab Version 13
Zugangsdaten	Öffentlicher Sitekey aus der Ansicht **Details**

5.5.2.2 SDK einbinden

Mit dem Swift Package Manager ergänzen Sie die Abhängigkeiten in `Package.swift`:

```
.package(url: "https://github.com/Myra-Security-GmbH/eu-captcha-ios-sdk.git",
from: "1.0.0")
```

In Xcode fügen Sie das Paket alternativ über **File > Add Package Dependencies** hinzu.

Mit CocoaPods ergänzen Sie die Datei `Podfile`:

```
pod 'EuCaptcha', '~> 1.0'
```

Führen Sie anschließend `pod install` aus.

5.5.2.3 Widget erzeugen

```
import EuCaptcha

let widget = EuCaptchaSDK.createWidget(
    sitekey: "EUCAPTCHA_SITE_KEY",
    theme: .auto,          // .light, .dark, or .auto
```

```
    language: "en"         // BCP 47 language code; defaults to system
language
)
```

Der Wert `theme` steuert die Darstellung mit `.light`, `.dark` und `.auto`. Der Wert `language` erwartet einen Sprachcode nach BCP 47. Ohne Angabe gilt die Sprache des Geräts.

5.5.2.4 Ereignisse auswerten

```
widget.onComplete = { event in
    // event.response is the token to verify server-side
    submitTokenToServer(event.response)
}

widget.onExpire = { _ in
    // The token has expired; the widget resets automatically
    disableLoginButton()
}

widget.onError = { _ in
    // An error occurred during the proof-of-work
    showErrorMessage()
}

widget.onStateChange = { event in
    print("State:", event.state, "Response:", event.response)
}
```

5.5.2.5 Widget einbetten

```
let widgetVC = widget.viewController()
addChild(widgetVC)
view.addSubview(widgetVC.view)
widgetVC.didMove(toParent: self)

widgetVC.view.translatesAutoresizingMaskIntoConstraints = false
NSLayoutConstraint.activate([
    widgetVC.view.leadingAnchor.constraint(equalTo: view.leadingAnchor,
constant: 16),
    widgetVC.view.trailingAnchor.constraint(equalTo: view.trailingAnchor,
constant: -16),
    widgetVC.view.heightAnchor.constraint(equalToConstant:
EuCaptchaSDK.preferredHeight),
])
```

5.5.2.6 Lebenszyklus des Widgets

Methode	Wirkung
`start()`	Ohne Wirkung. Das Widget startet selbstständig.
`reset()`	Verwirft den Zustand und startet eine neue Challenge.
`destroy()`	Gibt die Ressourcen frei. Verwenden Sie die Referenz danach nicht mehr.
`getResponse()`	Gibt das aktuelle Token zurück.
`getState()`	Gibt den aktuellen Zustand `EuCaptchaWidgetState` zurück.

5.5.2.7 Zustände des Widgets

Zustand	Bedeutung
`.initial`	Das Widget ist eingerichtet, aber noch nicht aktiv.
`.checking`	Die Challenge läuft.
`.completed`	Die Challenge ist bestanden. Das Feld `response` enthält das Token.
`.expired`	Das Token ist abgelaufen.
`.error`	Es ist ein Fehler aufgetreten.
`.destroyed`	Das Widget wurde abgebaut.

5.5.2.8 Token prüfen

Senden Sie den Wert `event.response` als `client_token` an Ihren Server. Der Server prüft das Token:

```
POST https://api.eu-captcha.eu/v1/verify
Content-Type: application/json

{
  "sitekey":           "EUCAPTCHA_SITE_KEY",
  "secret":            "EUCAPTCHA_SECRET_KEY",
  "client_ip":         "<client IP address>",
  "client_token":      "<token from the widget>",
```

```
  "client_user_agent": "<client user-agent>"
}
```

Die Antwort lautet `{ "success": true, "train": false }`.

Siehe **Endpunkt /verify**.

5.5.3 Flutter

Für Flutter bettet das Widget das EU CAPTCHA in eine WebView innerhalb der App ein und stellt eine Schnittstelle in Dart bereit.

5.5.3.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Flutter	ab Version 3.0
Dart SDK	ab Version 3.0
Betriebssystem	Android ab API-Ebene 19, iOS ab Version 13
Zugangsdaten	Öffentlicher Sitekey aus der Ansicht **Details**

5.5.3.2 Abhängigkeiten eintragen

Ergänzen Sie die Datei `pubspec.yaml`:

```
dependencies:
  flutter_inappwebview: ^6.0.0
  http: ^1.2.0
```

Kopieren Sie die Datei `lib/eucaptcha.dart` und das Verzeichnis `assets/` in Ihr Projekt und tragen Sie die Dateien ein:

```
flutter:
  assets:
    - assets/euc_check.all.js
    - assets/euc_styles.css
```

5.5.3.3 Plattformen einrichten

Ergänzen Sie unter Android die Datei `android/app/src/main/AndroidManifest.xml`:

```
<uses-permission android:name="android.permission.INTERNET"/>
```

Unter iOS ist keine weitere Einstellung erforderlich.

5.5.3.4 Widget einbinden

```
import 'eucaptcha.dart';

EuCaptcha(
  sitekey: 'EUCAPTCHA_SITE_KEY',
  theme: 'auto',   // 'light', 'dark', or 'auto'
  lang: 'en',      // BCP 47 language code
  onComplete: (response) {
    // response is the token — submit to your server as "eu-captcha-response"
  },
  onExpire: (_) { /* CAPTCHA expired */ },
  onError:  (_) { /* error occurred  */ },
)
```

Der Wert `theme` steuert die Darstellung mit `light`, `dark` und `auto`. Der Wert `lang` erwartet einen Sprachcode nach BCP 47.

5.5.3.5 Widget zurücksetzen

Verwenden Sie einen `GlobalKey<EuCaptchaState>`, um `reset()` aufzurufen:

```
final _captchaKey = GlobalKey<EuCaptchaState>();

EuCaptcha(key: _captchaKey, sitekey: '...', onComplete: ...)

// later:
_captchaKey.currentState?.reset();
```

5.5.3.6 Token prüfen

Senden Sie das Token nach dem Aufruf von `onComplete` an Ihren Server. Der Server prüft das Token:

```
POST https://api.eu-captcha.eu/v1/verify
Content-Type: application/json

{
  "sitekey":           "EUCAPTCHA_SITE_KEY",
  "secret":            "EUCAPTCHA_SECRET_KEY",
  "client_ip":         "<client IP>",
  "client_token":      "<token from widget>",
  "client_user_agent": "<user-agent>"
}
```

Die Antwort lautet `{ "success": true, "train": false }`.

Siehe **Endpunkt /verify**.

5.6 Infrastruktur

5.6.1 Caddy

Das Modul für Caddy stellt jeder Anfrage eine Challenge voran. Wer sie besteht, erhält ein signiertes Sitzungs-Cookie und erreicht Ihre Anwendung ohne weitere Unterbrechung, bis die Frist abläuft. Alle übrigen Anfragen leitet das Modul auf die Challenge-Seite um.

Der Schutz gilt damit für die gesamte Domain, nicht für einzelne Formulare.

5.6.1.1 Ablauf

1. Das Modul prüft jede Anfrage auf ein gültiges, an die IP-Adresse gebundenes Sitzungs-Cookie mit HMAC-Signatur.
2. Ohne gültiges Cookie leitet es die Anfrage auf den Pfad der Challenge um.
3. Nach der bestandenen Challenge prüft das Modul das Token serverseitig über die API.
4. Bei Erfolg setzt es das signierte Cookie und leitet auf die ursprüngliche Adresse zurück.

5.6.1.2 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Caddy	ab Version 2.9
Werkzeug	`xcaddy` für den Bau mit dem Modul
Go	ab Version 1.22 für den Bau aus den Quellen
Zugangsdaten	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

5.6.1.3 Caddy mit dem Modul bauen

Bauen Sie Caddy mit `xcaddy`:

```
xcaddy build --with github.com/Myra-Security-GmbH/eu-captcha-caddy
```

Aus den Quellen bauen Sie wie folgt:

```
git clone https://github.com/Myra-Security-GmbH/eu-captcha-caddy.git
cd eu-captcha-caddy
xcaddy build --with github.com/Myra-Security-GmbH/eu-captcha-caddy=.
```

5.6.1.4 Modul einrichten

Ergänzen Sie die Datei `Caddyfile` :

```
example.com {
    eu_captcha {
        sitekey        EUCAPTCHA_SITE_KEY
        secret         EUCAPTCHA_SECRET_KEY
        cookie_secret  <random-32-byte-hex>   # openssl rand -hex 32
        grace_period   1h
    }

    reverse_proxy localhost:3000
}
```

In der Konfiguration als JSON lautet der Eintrag:

```
{
  "handler": "eu_captcha",
  "sitekey": "EUCAPTCHA_SITE_KEY",
  "secret": "EUCAPTCHA_SECRET_KEY",
  "cookie_secret": "<random-32-byte-hex>",
  "grace_period": "1h"
}
```

5.6.1.5 Einstellungen

Die folgenden Einstellungen stehen zur Verfügung:

Einstellung	Pflicht	Vorgabe	Wirkung
`sitekey`	ja	–	Öffentlicher Sitekey.
`secret`	ja	–	Geheimer Schlüssel.

Einstellung	Pflicht	Vorgabe	Wirkung
`cookie_secret`	ja	–	Schlüssel für die HMAC-Signatur der Sitzungs-Cookies. Erzeugen Sie ihn mit `openssl rand -hex 32`.
`grace_period`	nein	`1h`	Gültigkeitsdauer einer bestandenen Challenge als Zeitangabe im Format von Go.
`cookie_name`	nein	`__eucaptcha`	Name des Sitzungs-Cookies.
`challenge_path`	nein	`/__captcha__`	Pfadanfang für die Challenge und für die Prüfung.
`verify_url`	nein	`https://api.eu-captcha.eu/v1/verify`	Adresse des Endpunkts für die Prüfung des Tokens.
`credentials_check_url`	nein	`https://api.eu-captcha.eu/v1/verify-credentials`	Adresse des Endpunkts für die Prüfung der Zugangsdaten. Das Modul ruft ihn beim Start auf.

5.6.1.6 Belegte Pfade

Das Modul belegt zwei Pfade Ihrer Domain. Verwenden Sie diese Pfade nicht in Ihrer Anwendung:

Pfad	Verwendung
`{challenge_path}`	Zeigt die Challenge-Seite an.
`{challenge_path}/verify`	Nimmt das gelöste Token per POST entgegen.

Mit der Vorgabe für `challenge_path` sind das die Pfade `/__captcha__` und `/__captcha__/verify`.

5.6.2 CrowdSec

Der Bouncer für CrowdSec ist ein eigenständiger vorgelagerter Server, der die Entscheidungen von CrowdSec mit dem Myra EU CAPTCHA verbindet. IP-Adressen mit der Entscheidung `captcha` erhalten eine Challenge, IP-Adressen mit der Entscheidung `ban` den Statuscode 403. Der übrige Verkehr erreicht Ihre Anwendung unverändert.

5.6.2.1 Ablauf

1. Der Bouncer fragt den Entscheidungsstrom von CrowdSec unter `/v1/decisions/stream` ab und hält die Entscheidungen im Arbeitsspeicher.
2. Bei jeder Anfrage schlägt er die IP-Adresse des Besuchers nach:

Entscheidung	Verhalten
`ban`	Der Bouncer beantwortet die Anfrage mit dem Statuscode 403 Forbidden.
`captcha`	Der Bouncer leitet auf `/__captcha__` um, zeigt das Widget an, prüft das Token serverseitig, setzt ein signiertes Cookie und leitet zurück.
keine Entscheidung	Der Bouncer reicht die Anfrage an Ihre Anwendung weiter.

5.6.2.2 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
CrowdSec	Laufender Agent mit erreichbarer Local API
Go	ab Version 1.22 für den Bau aus den Quellen
Zugangsdaten	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

5.6.2.3 Bouncer installieren

Laden Sie das fertige Programm für Ihre Plattform von der Seite der Veröffentlichungen herunter:

```
# Linux amd64
curl -L https://github.com/Myra-Security-GmbH/eu-captcha-crowdsec/releases/
latest/download/cs-eucaptcha-bouncer-linux-amd64 \
  -o cs-eucaptcha-bouncer
chmod +x cs-eucaptcha-bouncer
```

Programme stehen für Linux (amd64, arm64), macOS (amd64, arm64) und Windows (amd64) bereit.

Aus den Quellen bauen Sie wie folgt:

```
git clone https://github.com/Myra-Security-GmbH/eu-captcha-crowdsec.git
cd eu-captcha-crowdsec
make build
# produces ./cs-eucaptcha-bouncer
```

5.6.2.4 Bouncer einrichten

Um den Bouncer einzurichten, gehen Sie wie folgt vor:

- ▶ Melden Sie den Bouncer bei CrowdSec an:

```
cscli bouncers add eu-captcha-bouncer
# Note the printed API key — you'll need it in the next step
```

  - ↳ CrowdSec zeigt einen API-Schlüssel an.
- ▶ Legen Sie die Konfigurationsdatei an:

```
cp config.yaml.example config.yaml
```

► Tragen Sie die erforderlichen Werte ein:

```
listen_addr: "0.0.0.0:8080"
upstream_url: "http://localhost:3000"   # your application

crowdsec:
  lapi_url: "http://localhost:8080"
  api_key: "<API key from cscli bouncers add>"
  update_interval: "10s"

eu_captcha:
  sitekey: "EUCAPTCHA_SITE_KEY"    # from app.eu-captcha.eu
  secret:  "EUCAPTCHA_SECRET_KEY"  # from app.eu-captcha.eu

session:
  secret: "<random hex string>"  # openssl rand -hex 32
  ttl: "1h"
```

► Starten Sie den Bouncer:

```
./cs-eucaptcha-bouncer -config config.yaml
```

➔ Der Bouncer schreibt sein Protokoll als strukturiertes JSON auf die Standardausgabe.

Richten Sie anschließend Ihren Lastverteiler oder Ihr DNS auf die Adresse aus `listen_addr` aus.

5.6.2.5 Einstellungen

Die folgenden Einstellungen stehen zur Verfügung:

Einstellung	Vorgabe	Wirkung
`listen_addr`	`0.0.0.0:8080`	Adresse, auf der der Bouncer Anfragen entgegennimmt.
`upstream_url`	erforderlich	Adresse Ihrer Anwendung.
`crowdsec.lapi_url`	`http://localhost:8080`	Adresse der Local API von CrowdSec.
	erforderlich	API-Schlüssel aus `cscli bouncers add`.

Einstellung	Vorgabe	Wirkung
`crowdsec.api_key`		
`crowdsec.update_interval`	`10s`	Abstand zwischen zwei Abfragen des Entscheidungsstroms.
`eu_captcha.sitekey`	erforderlich	Öffentlicher Sitekey.
`eu_captcha.secret`	erforderlich	Geheimer Schlüssel.
`eu_captcha.verify_url`	`https://api.eu-captcha.eu/v1/verify`	Adresse des Endpunkts für die Prüfung.
`session.secret`	erforderlich	Schlüssel für die HMAC-Signatur. Erzeugen Sie ihn mit `openssl rand -hex 32`.
`session.cookie_name`	`__eucaptcha_pass`	Name des Sitzungs-Cookies.
`session.ttl`	`1h`	Gültigkeitsdauer einer bestandenen Challenge.
`trusted_proxies`	leer	CIDR-Bereiche, deren Kopfzeile `X-Forwarded-For` der Bouncer auswertet.

5.6.2.6 Betrieb über systemd

```
[Unit]
Description=EU Captcha CrowdSec Bouncer
After=network.target crowdsec.service

[Service]
ExecStart=/usr/local/bin/cs-eucaptcha-bouncer -config /etc/eu-captcha-bouncer/
config.yaml
Restart=on-failure
User=www-data

[Install]
WantedBy=multi-user.target
```

5.6.2.7 Betrieb im Container

```
FROM golang:1.22-alpine AS builder
WORKDIR /src
COPY . .
RUN go build -o cs-eucaptcha-bouncer ./cmd/cs-eucaptcha-bouncer

FROM alpine:3.19
COPY --from=builder /src/cs-eucaptcha-bouncer /usr/local/bin/
ENTRYPOINT ["cs-eucaptcha-bouncer", "-config", "/etc/bouncer/config.yaml"]
```

5.6.2.8 Belegte Pfade

Der Bouncer belegt zwei Pfade der geschützten Domain. Verwenden Sie diese Pfade nicht in Ihrer Anwendung:

Pfad	Verwendung
`/__captcha__`	Zeigt die Challenge-Seite an.
`/__captcha__/verify`	Nimmt das gelöste Token entgegen.

5.6.3 Traefik

Traefik bindet das Myra EU CAPTCHA über das Plugin **CrowdSec Bouncer Traefik Plugin** ein. Das Plugin stammt nicht von Myra Security. Es fragt die Entscheidungen von CrowdSec ab und zeigt für die Entscheidung `captcha` eine Challenge des Myra EU CAPTCHA an.

5.6.3.1 Ablauf

Das Plugin arbeitet als Middleware vor Ihrem Dienst:

Entscheidung	Verhalten
`ban`	Das Plugin weist die Anfrage ab.
`captcha`	Das Plugin zeigt die Vorlage `captcha.html` mit dem Widget an, prüft das gelöste Token und lässt die IP-Adresse für die Dauer der Schonfrist durch.
keine Entscheidung	Das Plugin reicht die Anfrage unverändert an Ihren Dienst weiter.

Anders als bei **Caddy** entscheidet damit CrowdSec, wer eine Challenge sieht. Alle Anfragen einer Domain schützen Sie mit dem eigenständigen Bouncer. Siehe **CrowdSec**.

5.6.3.2 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Traefik	Version, die Plugins unterstützt
CrowdSec	Laufender Agent mit erreichbarer Local API
Zugangsdaten	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

5.6.3.3 Plugin registrieren

Um das Plugin bekannt zu machen, gehen Sie wie folgt vor:

- ▶ Ergänzen Sie die statische Konfiguration `traefik.yml`:

```
experimental:
  plugins:
    bouncer:
      moduleName: github.com/maxlerebourg/crowdsec-bouncer-traefik-plugin
      version: v1.3.5
```

- ▶ Starten Sie Traefik neu.
- → Traefik lädt das Plugin beim Start herunter.

5.6.3.4 Vorlage und Bouncer vorbereiten

Um die Vorlage der Challenge bereitzustellen und den Bouncer anzumelden, gehen Sie wie folgt vor:

- ▶ Laden Sie die Vorlage herunter:

```
curl -o captcha.html \
  https://raw.githubusercontent.com/maxlerebourg/crowdsec-bouncer-traefik-
plugin/main/captcha.html
```

- ▶ Binden Sie die Datei als Volume in den Container von Traefik ein.
- ▶ Melden Sie den Bouncer bei CrowdSec an:

```
docker exec crowdsec cscli bouncers add traefik-bouncer
```

  - ↳ Der Befehl gibt einen API-Schlüssel aus. Notieren Sie ihn.

5.6.3.5 Middleware einrichten

Um die Middleware einzurichten, ergänzen Sie die dynamische Konfiguration:

```
http:
  middlewares:
```

```
    crowdsec:
      plugin:
        bouncer:
          crowdsecLapiKey: "<API-Schlüssel aus cscli bouncers add>"
          crowdsecLapiHost: "crowdsec:8080"
          captchaProvider: eucaptcha
          captchaSiteKey: "EUCAPTCHA_SITE_KEY"
          captchaSecretKey: "EUCAPTCHA_SECRET_KEY"
          captchaGracePeriodSeconds: 1800
          captchaHTMLFilePath: /captcha.html
```

Die folgenden Schlüssel betreffen das Myra EU CAPTCHA:

Schlüssel	Beschreibung
`captchaProvider`	Anbieter der Challenge. Für das Myra EU CAPTCHA lautet der Wert `eucaptcha` .
`captchaSiteKey`	Öffentlicher Sitekey.
`captchaSecretKey`	Secret des Sitekeys.
`captchaGracePeriodSeconds`	Dauer in Sekunden, für die eine bestandene Challenge gilt.
`captchaHTMLFilePath`	Pfad zur Vorlage `captcha.html` im Container.

- ► Weisen Sie die Middleware den Routen zu, die geschützt werden sollen.

Das Secret gehört ausschließlich in die Konfiguration auf dem Server. Hinterlegen Sie es in einer Umgebungsvariablen oder in einem Dienst zur Verwaltung von Geheimnissen.

Dieselben Schritte zeigt auch der Installationsassistent im Reiter **Integration** des Sitekeys. Siehe **Integration**.

5.7 Beispiele

5.7.1 React und PHP

Dieses Beispiel schützt das Kontaktformular einer bestehenden Anwendung. Das Frontend ist eine React-Anwendung mit Vite, das Backend ein PHP-Server. Beide laufen bereits. Für das Myra EU CAPTCHA ändern Sie nur die unten mit einem Pfeil markierten Dateien.

5.7.1.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Node.js mit npm	installiert
PHP	ab Version 8.0
Composer	installiert
Sitekey	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

Der öffentliche Sitekey gehört ins Frontend, das Secret ausschließlich ins Backend. Ein Secret im Auslieferungspaket des Frontends liest jeder Besucher mit.

5.7.1.2 Projektstruktur

Die folgenden Dateien ändern sich. Alles Übrige im Projekt bleibt unverändert:

```
mein-projekt/
├── frontend/                    # bestehende React-Anwendung (Vite)
│   ├── package.json             # ← Schritt 1: Paket @myrasec/eu-captcha
eintragen
│   ├── vite.config.ts
│   ├── index.html
│   └── src/
│       ├── main.tsx
│       ├── App.tsx
│       └── ContactForm.tsx      # ← Schritt 1 und 2: Widget einbinden, Token
```

```
absenden
└── backend/                        # bestehender PHP-Server
    ├── composer.json               # ← Schritt 3: Paket myra-security-gmbh/eu-
captcha eintragen
    ├── vendor/
    ├── .env                        # ← Sitekey und Secret eintragen
    └── public/
        └── index.php               # ← Schritt 3: Prüfung an den Anfang des
Handlers setzen
```

5.7.1.3 Schritt 1: Widget einbinden

Das Formular steht in der Datei `frontend/src/ContactForm.tsx` .

Installieren Sie das Paket:

```
npm i @myrasec/eu-captcha
```

Setzen Sie die Komponente `EuCaptcha` in das Formular und nehmen Sie das Token über den Callback `onComplete` entgegen:

```
import { useState } from "react";
import { EuCaptcha, isEuCaptchaDone } from "@myrasec/eu-captcha";

const captchaSitekey = "EUCAPTCHA_SITE_KEY";

export function ContactForm() {
  const [token, setToken] = useState("");

  // handleSubmit folgt in Schritt 2

  return (
    <form onSubmit={handleSubmit}>
      <input name="email" type="email" required />
      <textarea name="message" required />


{/* onComplete liefert das Token, sobald die Challenge abgeschlossen ist */}
      <EuCaptcha sitekey={captchaSitekey} onComplete={setToken} />

      <button type="submit">Submit</button>
    </form>
  );
}
```

Die Komponente lädt das Skript `verify.js`, legt den versteckten iframe an und startet die Challenge selbsttätig.

5.7.1.4 Schritt 2: Token absenden

Prüfen Sie vor dem Absenden mit `isEuCaptchaDone()`, ob die Challenge abgeschlossen ist. Senden Sie das Token anschließend im Feld `eu-captcha-response` des JSON-Rumpfs:

```
async function handleSubmit(e: React.FormEvent<HTMLFormElement>):
Promise<void> {
  e.preventDefault();
  if (!isEuCaptchaDone()) {
    // challenge not yet complete
    return;
  }
  const fields = Object.fromEntries(new FormData(e.currentTarget));
  const res = await fetch("/api/submit", {
    method: "POST",
    headers: { "Content-Type": "application/json" },
    body: JSON.stringify({ ...fields, "eu-captcha-response": token }),
  });
  if (res.ok) {
    // success
  }
}
```

Die Komponente `EuCaptcha` ist mit `memo` umhüllt und zeichnet deshalb nicht bei jeder Eingabe im Formular neu. Die Optimierung greift nur bei gleichbleibenden Referenzen. Die Funktion `setToken` aus `useState` bleibt gleich. Eigene Inline-Funktionen für `onComplete`, `onExpired` oder `onError` kapseln Sie mit `useCallback`.

5.7.1.5 Schritt 3: Token prüfen

Die Prüfung gehört in den serverseitigen Handler, der die Formulardaten entgegennimmt, hier `/api/submit` in der Datei `backend/public/index.php`. Sie steht an erster Stelle, vor jeder Verarbeitung.

Installieren Sie das Paket:

```
composer require myra-security-gmbh/eu-captcha
```

Lesen Sie die Felder aus dem JSON-Rumpf und setzen Sie die Prüfung an den Anfang des Handlers:

```php
<?php

require __DIR__ . '/vendor/autoload.php';

use Myrasec\EuCaptcha;

// Felder aus dem JSON-Rumpf lesen
$body = json_decode(file_get_contents('php://input'), true) ?: [];

// Prüfung vor jeder Verarbeitung
$captcha = new EuCaptcha(
    sitekey: getenv('EUCAPTCHA_SITE_KEY'),
    secret:  getenv('EUCAPTCHA_SECRET_KEY'),
);

if (!$captcha->validate()->success()) {
    http_response_code(400);
    exit('captcha verification failed');
}

// ab hier die bisherige Logik
$email   = $body['email'];
$message = $body['message'];
mail('team@example.com', 'Kontakt', $message);
echo 'Danke!';
```

Der Aufruf `validate()` liest das Token selbsttätig aus dem Feld `eu-captcha-response`, sowohl aus einem Formularfeld als auch aus einem JSON-Rumpf. Die IP-Adresse des Besuchers liest der Aufruf aus den Kopfzeilen. Halten Sie Sitekey und Secret in der Datei `.env`, aus der `getenv` sie liest.

Siehe **PHP**.

5.7.1.6 Integration prüfen

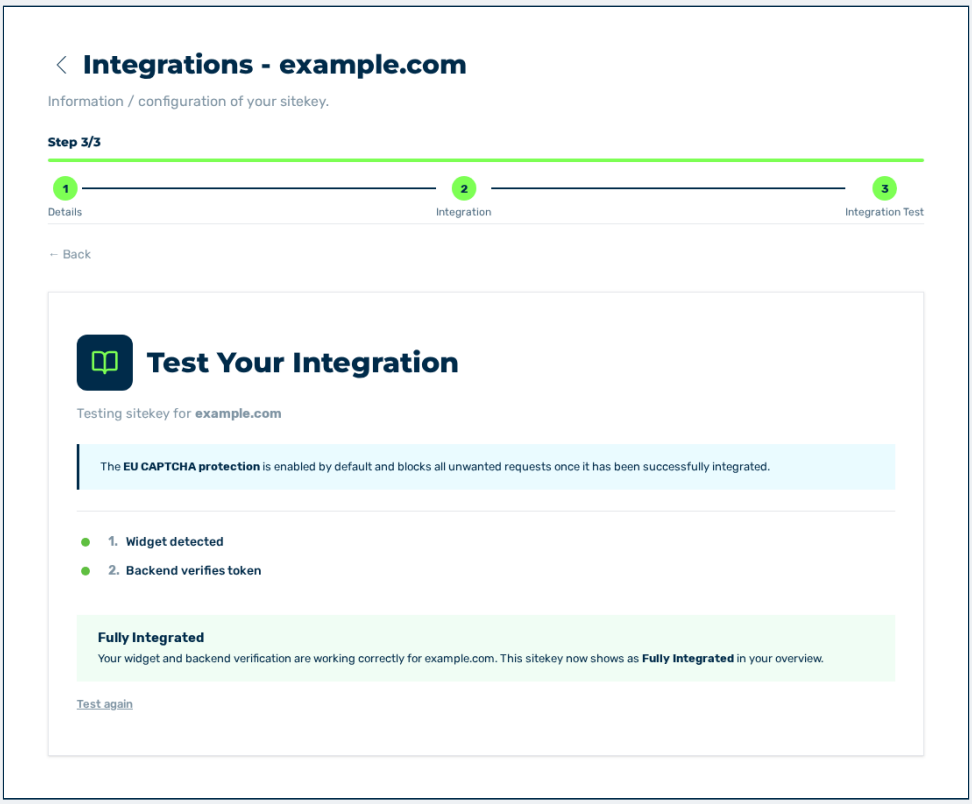


Abbildung 150: Ansicht Integration Test mit dem Ergebnis Fully Integrated

Prüfen Sie zum Schluss in der Ansicht **Integration Test** des Sitekeys, ob Frontend und Backend zusammenspielen. Danach trägt der Sitekey den Status **Fully Integrated**.

Siehe **Integration testen**.

5.7.2 Vue und Python

Dieses Beispiel schützt das Kontaktformular einer bestehenden Anwendung. Das Frontend ist eine Vue-Anwendung mit Vite, das Backend ein Python-Server mit Flask. Beide laufen bereits. Für das Myra EU CAPTCHA ändern Sie nur die unten mit einem Pfeil markierten Dateien.

5.7.2.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Node.js mit npm	installiert
Python	Version 3 mit pip
Sitekey	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

Der öffentliche Sitekey gehört ins Frontend, das Secret ausschließlich ins Backend. Ein Secret im Auslieferungspaket des Frontends liest jeder Besucher mit.

5.7.2.2 Projektstruktur

Die folgenden Dateien ändern sich. Alles Übrige im Projekt bleibt unverändert:

```
mein-projekt/
├── frontend/                    # bestehende Vue-Anwendung (Vite)
│   ├── package.json             # ← Schritt 1: Paket @myrasec/eu-captcha-vue
eintragen
│   ├── vite.config.js
│   ├── index.html
│   └── src/
│       ├── main.js
│       ├── App.vue
│       └── ContactForm.vue      # ← Schritt 1 und 2: Widget einbinden, Token
absenden
└── backend/                     # bestehender Python-Server (Flask)
    ├── .venv/
    ├── requirements.txt         # ← Schritt 3: Paket myra-eucaptcha
eintragen
```

```
    └── app.py                    # ← Schritt 3: Prüfung an den Anfang der
Route setzen
```

5.7.2.3 Schritt 1: Widget einbinden

Das Formular steht in der Datei `frontend/src/ContactForm.vue` .

Installieren Sie das Paket:

```
npm i @myrasec/eu-captcha-vue
```

Setzen Sie die Komponente `EuCaptcha` in das Formular und nehmen Sie das Token über den Callback `onComplete` entgegen:

```
<template>
  <form @submit.prevent="handleSubmit">
    <input name="email" type="email" required />
    <textarea name="message" required></textarea>

    <!-- onComplete liefert das Token, sobald die Challenge abgeschlossen ist
-->
    <EuCaptcha :sitekey="captchaSitekey" :onComplete="onComplete" />

    <button type="submit">Submit</button>
  </form>
</template>
```

Die Komponente lädt das Skript `verify.js` , legt den versteckten iframe an und startet die Challenge selbsttätig.

5.7.2.4 Schritt 2: Token absenden

Prüfen Sie vor dem Absenden mit `isEuCaptchaDone()` , ob die Challenge abgeschlossen ist. Senden Sie das Token anschließend im Feld `eu-captcha-response` des JSON-Rumpfs:

```
<script setup>
import { ref } from "vue";
import { EuCaptcha, isEuCaptchaDone } from "@myrasec/eu-captcha-vue";

const captchaSitekey = "EUCAPTCHA_SITE_KEY";
const token = ref("");
```

```
function onComplete(t) {
  token.value = t;
}

async function handleSubmit(e) {
  if (!isEuCaptchaDone()) {
    // challenge not yet complete
    return;
  }
  const fields = Object.fromEntries(new FormData(e.target));
  const res = await fetch("/api/contact", {
    method: "POST",
    headers: { "Content-Type": "application/json" },
    body: JSON.stringify({ ...fields, "eu-captcha-response": token.value }),
  });
  if (res.ok) {
    // success
  }
}
</script>
```

5.7.2.5 Schritt 3: Token prüfen

Die Prüfung gehört in den serverseitigen Handler, der die Formulardaten entgegennimmt, hier die Route `/api/contact` in der Datei `backend/app.py`. Sie steht an erster Stelle, vor jeder Verarbeitung.

Installieren Sie das Paket:

```
pip install myra-eucaptcha
```

Lesen Sie die Felder aus dem JSON-Rumpf und setzen Sie die Prüfung an den Anfang der Route:

```
import os

from flask import Flask, abort, request
from myra_eucaptcha import MyraEuCaptchaClient, MyraEuCaptchaClientConfig

app = Flask(__name__)

client = MyraEuCaptchaClient(config=MyraEuCaptchaClientConfig(
    sitekey=os.environ["EUCAPTCHA_SITE_KEY"],
    secret=os.environ["EUCAPTCHA_SECRET_KEY"],
))
```

```python
@app.post("/api/contact")
def contact():
    body = request.get_json(silent=True) or {}

    # Prüfung vor jeder Verarbeitung
    result = client.validate(
        token=body.get("eu-captcha-response", ""),
        remote_addr=request.headers.get("x-real-ip", ""),
    )
    if not result.success:
        abort(400, "captcha verification failed")

    # ab hier die bisherige Logik
    email = body.get("email")
    message = body.get("message")
    return "Danke!"
```

Der Python-Client verlangt das Token ausdrücklich. Lesen Sie es aus dem Feld `eu-captcha-response` des JSON-Rumpfs und übergeben Sie es zusammen mit der IP-Adresse des Besuchers an `validate()`. Halten Sie Sitekey und Secret in Umgebungsvariablen, aus denen `os.environ` sie liest.

Im Trainingsmodus liefert `result.success` den Wert `True`, und die Übermittlung läuft durch. Der Trainingsmodus greift bei einem unbekannten Sitekey, bei einem falschen Secret und bei abgeschaltetem Schutz. Prüfen Sie deshalb immer mit den echten Zugangsdaten aus dem Kundenportal.

Siehe **Python**.

5.7.2.6 Integration prüfen

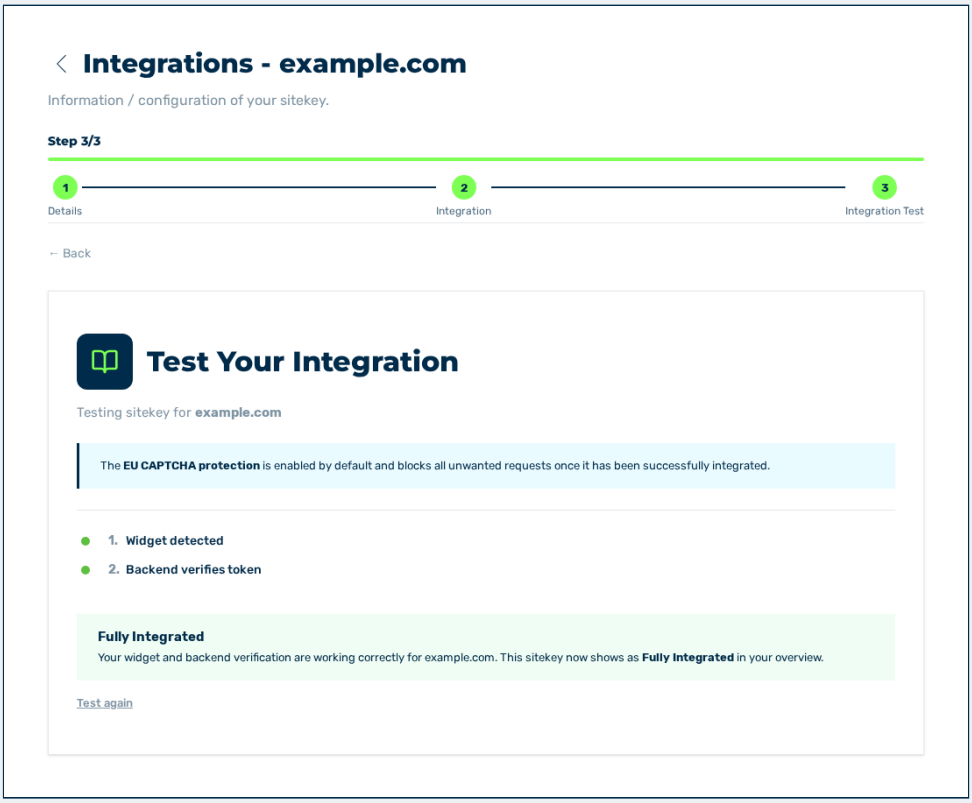


Abbildung 151: Ansicht Integration Test mit dem Ergebnis Fully Integrated

Prüfen Sie zum Schluss in der Ansicht **Integration Test** des Sitekeys, ob Frontend und Backend zusammenspielen. Danach trägt der Sitekey den Status **Fully Integrated**.

Siehe **Integration testen**.

5.7.3 Angular und Java

Dieses Beispiel schützt das Kontaktformular einer bestehenden Anwendung. Das Frontend ist eine Angular-Anwendung, das Backend ein Java-Server mit Spring Boot. Beide laufen bereits. Für das Myra EU CAPTCHA ändern Sie nur die unten mit einem Pfeil markierten Dateien.

5.7.3.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Node.js mit npm	installiert
JDK	Version 17 mit Maven oder Gradle
Sitekey	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

Der öffentliche Sitekey gehört ins Frontend, das Secret ausschließlich ins Backend. Ein Secret im Auslieferungspaket des Frontends liest jeder Besucher mit.

5.7.3.2 Projektstruktur

Die folgenden Dateien ändern sich. Alles Übrige im Projekt bleibt unverändert:

```
mein-projekt/
├── frontend/                          # bestehende Angular-Anwendung
│   ├── package.json                   # ← Schritt 1: Paket @myrasec/eu-
captcha-angular20 eintragen
│   ├── angular.json
│   └── src/app/
│       ├── app.component.ts
│       └── contact-form.component.ts  # ← Schritt 1 und 2: Widget
einbinden, Token absenden
└── backend/                           # bestehender Java-Server (Spring
Boot)
    ├── pom.xml                        # ← Schritt 3: Abhängigkeit eintragen
    └── src/main/java/com/beispiel/
        ├── Application.java
        └── ContactController.java     # ← Schritt 3: Prüfung an den Anfang
der Methode setzen
```

5.7.3.3 Schritt 1: Widget einbinden

Das Formular steht in der Datei `frontend/src/app/contact-form.component.ts` .

Installieren Sie das Paket zu Ihrer Angular-Version, hier Angular 20:

```
npm i @myrasec/eu-captcha-angular20
```

Setzen Sie die Komponente in das Formular und nehmen Sie das Token über das Ereignis `completed` entgegen:

```
import { Component } from '@angular/core';
import { EuCaptchaComponent, isEuCaptchaDone } from '@myrasec/eu-captcha-
angular20';

@Component({
  standalone: true,
  imports: [EuCaptchaComponent],
  template: `
    <form (submit)="handleSubmit($event)">
      <input name="email" type="email" required />
      <textarea name="message" required></textarea>

      <!-- (completed) liefert das Token, sobald die Challenge abgeschlossen
ist -->
      <eu-captcha sitekey="EUCAPTCHA_SITE_KEY" (completed)="token = $event" />

      <button type="submit">Submit</button>
    </form>
  `,
})
export class ContactFormComponent {
  token = '';
  // handleSubmit folgt in Schritt 2
}
```

Der Selektor `<eu-captcha>` lädt das Skript `verify.js` , legt den versteckten iframe an und startet die Challenge selbsttätig.

Eine Übersicht der Pakete je Angular-Version steht unter **Angular**.

5.7.3.4 Schritt 2: Token absenden

Prüfen Sie vor dem Absenden mit `isEuCaptchaDone()`, ob die Challenge abgeschlossen ist. Senden Sie das Token anschließend im Feld `eu-captcha-response` des JSON-Rumpfs:

```
async handleSubmit(event: Event): Promise<void> {
  event.preventDefault();
  if (!isEuCaptchaDone()) {
    // challenge not yet complete
    return;
  }
  const fields = Object.fromEntries(new FormData(event.target as
HTMLFormElement));
  const res = await fetch('/api/contact', {
    method: 'POST',
    headers: { 'Content-Type': 'application/json' },
    body: JSON.stringify({ ...fields, 'eu-captcha-response': this.token }),
  });
  if (res.ok) {
    // success
  }
}
```

5.7.3.5 Schritt 3: Token prüfen

Die Prüfung gehört in den serverseitigen Handler, der die Formulardaten entgegennimmt, hier die Methode zur Route `/api/contact` in der Datei `backend/src/main/java/com/beispiel/ContactController.java`. Sie steht an erster Stelle, vor jeder Verarbeitung.

Tragen Sie die Abhängigkeit in die Datei `pom.xml` ein:

```
<dependency>
  <groupId>com.myrasec</groupId>
  <artifactId>eu-captcha-java-resttemplate</artifactId>
  <version>1.0.0</version>
  <scope>compile</scope>
</dependency>
```

Lesen Sie den JSON-Rumpf als `Map` und setzen Sie die Prüfung an den Anfang der Methode:

```
import com.myrasec.client.ApiClient;
import com.myrasec.client.api.EuCaptchaApi;
import com.myrasec.client.model.VerifyRequest;
import com.myrasec.client.model.VerifyResponse;

@PostMapping("/api/contact")
public ResponseEntity<String> contact(
        @RequestBody Map<String, String> body,
        HttpServletRequest req) {

    // Prüfung vor jeder Verarbeitung
    EuCaptchaApi api = new EuCaptchaApi(new ApiClient());

    VerifyRequest request = new VerifyRequest()
            .sitekey(System.getenv("EUCAPTCHA_SITE_KEY"))
            .secret(System.getenv("EUCAPTCHA_SECRET_KEY"))
            .clientIp(req.getRemoteAddr())
            .clientToken(body.get("eu-captcha-response"))
            .clientUserAgent(req.getHeader("User-Agent"));

    VerifyResponse response = api.verifyClientToken(request);
    if (response.isTrainingMode() || !
Boolean.TRUE.equals(response.getSuccess())) {
        return ResponseEntity.badRequest().body("captcha verification
failed");
    }

    // ab hier die bisherige Logik
    String email = body.get("email");
    String message = body.get("message");
    return ResponseEntity.ok("Danke!");
}
```

Der Wert des Felds `eu-captcha-response` geht als `clientToken` in die Anfrage. Halten Sie Sitekey und Secret in Umgebungsvariablen, aus denen `System.getenv` sie liest.

Der Trainingsmodus greift bei einem unbekannten Sitekey, bei einem falschen Secret und bei einem Sitekey mit der Einstellung `train`. Das Beispiel weist die Übermittlung in diesem Fall zurück. So läuft eine Fehlkonfiguration nicht ungeprüft durch.

Siehe Java.

5.7.3.6 Integration prüfen

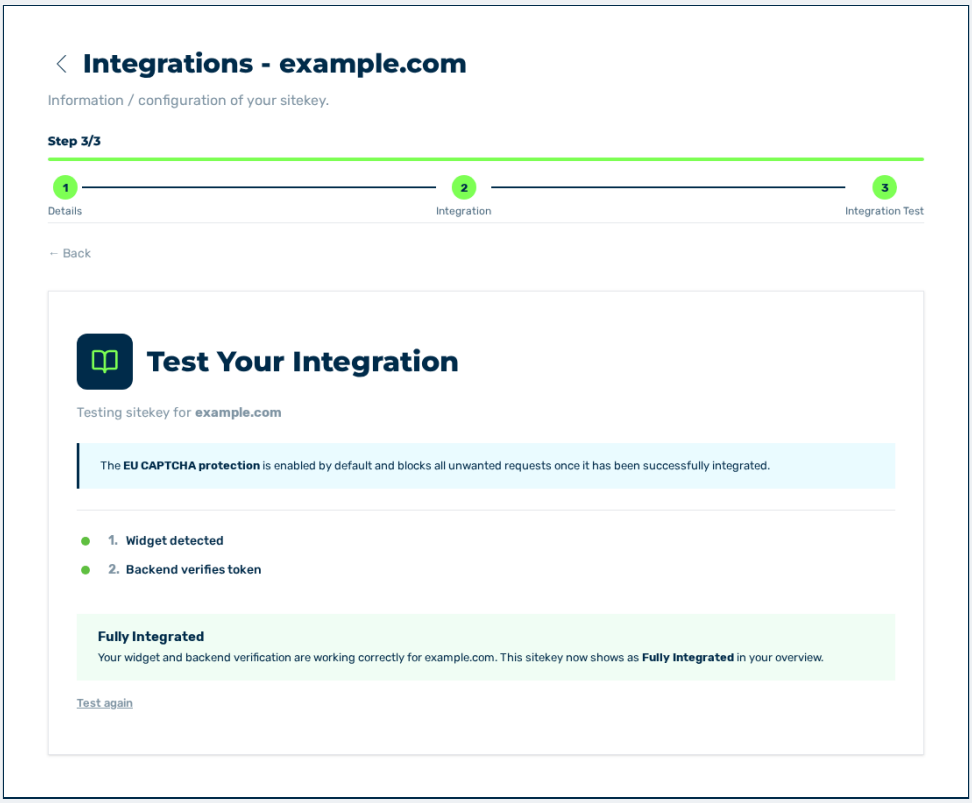


Abbildung 152: Ansicht Integration Test mit dem Ergebnis Fully Integrated

Prüfen Sie zum Schluss in der Ansicht **Integration Test** des Sitekeys, ob Frontend und Backend zusammenspielen. Danach trägt der Sitekey den Status **Fully Integrated**.

Siehe Integration testen.

5.7.4 Svelte und Ruby

Dieses Beispiel schützt das Kontaktformular einer bestehenden Anwendung. Das Frontend ist eine Svelte-Anwendung mit Vite, das Backend ein Ruby-Server mit Sinatra. Beide laufen bereits. Für das Myra EU CAPTCHA ändern Sie nur die unten mit einem Pfeil markierten Dateien.

5.7.4.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Node.js mit npm	installiert
Svelte	Version 5
Ruby	ab Version 2.7
Sitekey	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

Der öffentliche Sitekey gehört ins Frontend, das Secret ausschließlich ins Backend. Ein Secret im Auslieferungspaket des Frontends liest jeder Besucher mit.

5.7.4.2 Projektstruktur

Die folgenden Dateien ändern sich. Alles Übrige im Projekt bleibt unverändert:

```
mein-projekt/
├── frontend/                    # bestehende Svelte-Anwendung (Vite)
│   ├── package.json             # ← Schritt 1: Paket @myrasec/eu-captcha-
svelte eintragen
│   ├── vite.config.js
│   └── src/
│       ├── main.js
│       ├── App.svelte
│       └── ContactForm.svelte   # ← Schritt 1 und 2: Widget einbinden, Token
absenden
└── backend/                     # bestehender Ruby-Server (Sinatra)
    ├── Gemfile                  # ← Schritt 3: Gem eu_captcha eintragen
    └── app.rb                   # ← Schritt 3: Prüfung an den Anfang der
Route setzen
```

5.7.4.3 Schritt 1: Widget einbinden

Das Formular steht in der Datei `frontend/src/ContactForm.svelte`.

Installieren Sie das Paket:

```
npm i @myrasec/eu-captcha-svelte
```

Setzen Sie die Komponente `EuCaptcha` in das Formular und nehmen Sie das Token über den Callback `onComplete` entgegen:

```
<form onsubmit={handleSubmit}>
  <input name="email" type="email" required />
  <textarea name="message" required></textarea>

  <!-- onComplete liefert das Token, sobald die Challenge abgeschlossen ist --
>
  <EuCaptcha sitekey={captchaSitekey} onComplete={(t) => (token = t)} />

  <button type="submit">Submit</button>
</form>
```

Die Komponente lädt das Skript `verify.js`, legt den versteckten iframe an und startet die Challenge selbsttätig.

5.7.4.4 Schritt 2: Token absenden

Prüfen Sie vor dem Absenden mit `isEuCaptchaDone()`, ob die Challenge abgeschlossen ist. Senden Sie das Token anschließend im Feld `eu-captcha-response` des JSON-Rumpfs:

```
<script>
  import { EuCaptcha, isEuCaptchaDone } from "@myrasec/eu-captcha-svelte";

  const captchaSitekey = "EUCAPTCHA_SITE_KEY";
  let token = "";

  async function handleSubmit(e) {
    e.preventDefault();
    if (!isEuCaptchaDone()) {
      // challenge not yet complete
      return;
    }
    const fields = Object.fromEntries(new FormData(e.target));
```

```
    const res = await fetch("/api/contact", {
      method: "POST",
      headers: { "Content-Type": "application/json" },
      body: JSON.stringify({ ...fields, "eu-captcha-response": token }),
    });
    if (res.ok) {
      // success
    }
  }
</script>
```

5.7.4.5 Schritt 3: Token prüfen

Die Prüfung gehört in den serverseitigen Handler, der die Formulardaten entgegennimmt, hier die Route `/api/contact` in der Datei `backend/app.rb`. Sie steht an erster Stelle, vor jeder Verarbeitung.

Installieren Sie das Gem:

```
gem install eu_captcha
```

Lesen Sie die Felder aus dem JSON-Rumpf und setzen Sie die Prüfung an den Anfang der Route:

```
require 'eu_captcha'
require 'json'

captcha = EuCaptcha::Client.new(
  sitekey: ENV['EUCAPTCHA_SITE_KEY'],
  secret:  ENV['EUCAPTCHA_SECRET_KEY']
)

post '/api/contact' do
  payload = JSON.parse(request.body.read)

  # Prüfung vor jeder Verarbeitung
  result = captcha.validate(
    token:       payload['eu-captcha-response'],
    remote_addr: request.ip
  )
  halt 400, 'captcha verification failed' unless result.success?

  # ab hier die bisherige Logik
  email   = payload['email']
  message = payload['message']
```

```
  'Danke!'
end
```

Der Ruby-Client verlangt das Token ausdrücklich. Lesen Sie es aus dem Feld `eu-captcha-response` des JSON-Rumpfs und übergeben Sie es zusammen mit der IP-Adresse des Besuchers an `validate`. Halten Sie Sitekey und Secret in Umgebungsvariablen, aus denen `ENV` sie liest.

Die Methode `success?` liefert im Trainingsmodus den Wert `false`. Eine Fehlkonfiguration läuft damit nicht ungeprüft durch.

Siehe **Ruby**.

5.7.4.6 Integration prüfen

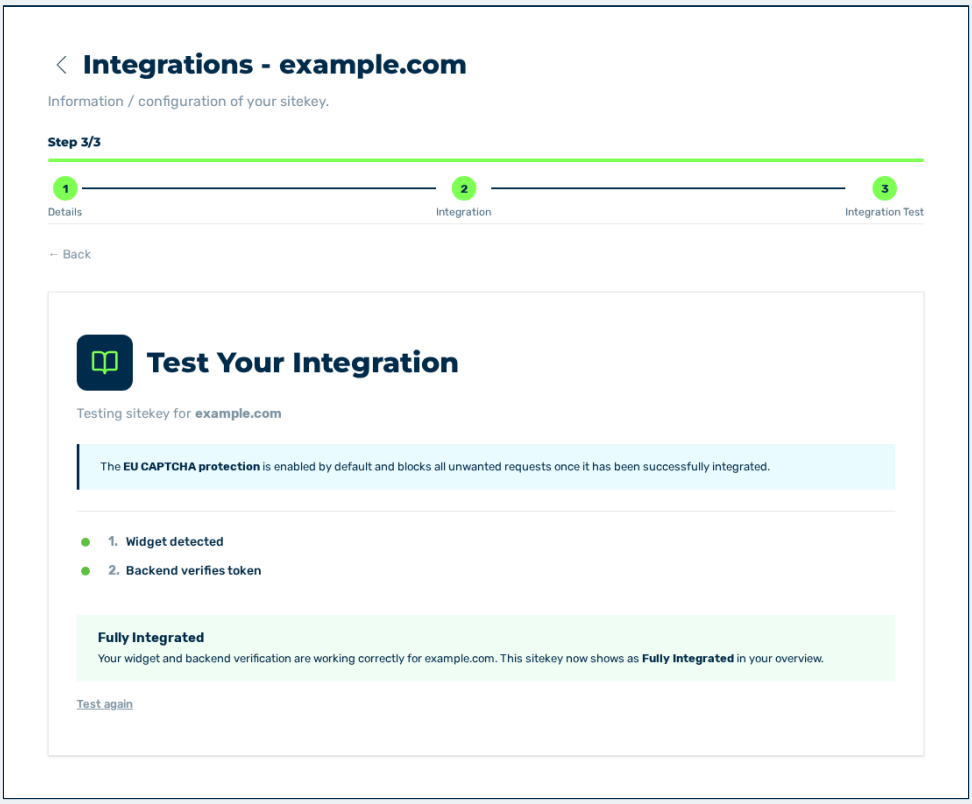


Abbildung 153: Ansicht Integration Test mit dem Ergebnis Fully Integrated

Prüfen Sie zum Schluss in der Ansicht **Integration Test** des Sitekeys, ob Frontend und Backend zusammenspielen. Danach trägt der Sitekey den Status **Fully Integrated**.

Siehe **Integration testen**.

5.7.5 HTML und Django

Dieses Beispiel schützt das Kontaktformular einer bestehenden Anwendung ohne Frontend-Framework. Die Seite wird als Django-Template ausgeliefert, das Formular sendet einen gewöhnlichen POST. Das Widget legt dabei selbst das versteckte Feld `eu-captcha-response` an, das mit dem Formular mitgeht.

> In einer Single-Page-Anwendung nehmen Sie das Token stattdessen über einen Callback entgegen und senden es als JSON. Siehe **React und PHP**.

5.7.5.1 Voraussetzungen

Die folgenden Voraussetzungen müssen erfüllt sein:

Voraussetzung	Wert
Django-Projekt	läuft und liefert die Seite aus
Python	Version 3 mit pip
Sitekey	Öffentlicher Sitekey und Secret aus der Ansicht **Details**

> Der öffentliche Sitekey gehört in die Seite, das Secret ausschließlich auf den Server. Ein Secret in einem Template liest jeder Besucher mit.

5.7.5.2 Projektstruktur

Die folgenden Dateien ändern sich. Alles Übrige im Projekt bleibt unverändert:

```
mein-projekt/                    # bestehendes Django-Projekt
├── manage.py
├── templates/
│   └── kontakt.html             # ← Schritt 1 und 2: Skript und Widget
einbinden
└── server/
    ├── settings.py              # ← Schritt 3: App und Zugangsdaten
eintragen
    ├── urls.py                  # ← Schritt 3: Route /kontakt/ registrieren
    └── views.py                 # ← Schritt 3: Prüfung an den Anfang der
View setzen
```

5.7.5.3 Schritt 1: Widget einbinden

Die Seite steht in der Datei `templates/kontakt.html` .

Laden Sie das Skript `verify.js` vom Myra-CDN und setzen Sie das Widget-Element in das Formular:

```
<script src="https://cdn.eu-captcha.eu/verify.js" async defer></script>

<form action="/kontakt/" method="post">
  <input name="email" type="email" required />
  <textarea name="message" required></textarea>

  <div class="eu-captcha" data-sitekey="EUCAPTCHA_SITE_KEY"></div>

  <button type="submit">Submit</button>
</form>
```

Das Skript legt den versteckten iframe an, startet die Challenge selbsttätig und fügt das versteckte Feld `eu-captcha-response` in das Formular ein. Das Feld geht beim Absenden mit.

5.7.5.4 Schritt 2: Absenden freigeben

Falls erforderlich, sperren Sie die Schaltfläche und geben sie erst nach der Challenge frei. Werten Sie dazu die Fensternachricht `euCaptchaCompleted` aus. Prüfen Sie dabei den Ursprung der Nachricht, da jedes Skript und jede Erweiterung im Browser Nachrichten senden kann:

```
<script>
  const CAPTCHA_ORIGIN = "https://cdn.eu-captcha.eu";
  const btn = document.querySelector('button[type="submit"]');
  btn.disabled = true;

  window.addEventListener("message", (msg) => {
    if (msg.origin !== CAPTCHA_ORIGIN) return;
    const data = msg.data ?? {};
    if (data.type === "euCaptchaCompleted") {
      btn.disabled = false;
    }
  });
</script>
```

> Die Sperre im Browser steuert nur die Bedienung. Die Prüfung des Tokens auf dem Server bleibt in jedem Fall erforderlich.

5.7.5.5 Schritt 3: Token prüfen

Die Prüfung gehört in die View, die den POST des Formulars entgegennimmt, hier die Route `/kontakt/` in der Datei `server/views.py`. Sie steht an erster Stelle, vor jeder Verarbeitung.

Installieren Sie die App:

```
pip install myra-eucaptcha-django
```

Tragen Sie die App und die Zugangsdaten in die Datei `settings.py` ein:

```
import os

INSTALLED_APPS = [
    # ...
    "myra_eucaptcha_django",
]

EUCAPTCHA_SITEKEY = os.environ["EUCAPTCHA_SITE_KEY"]
EUCAPTCHA_SECRET = os.environ["EUCAPTCHA_SECRET_KEY"]
```

Legen Sie die Tabellen der App an:

```
python manage.py migrate
```

Setzen Sie die Prüfung an den Anfang der View:

```
from django.core.exceptions import ValidationError
from django.http import HttpResponse, HttpResponseBadRequest
from myra_eucaptcha_django import validate_captcha


def kontakt(request):
    if request.method == "POST":
        # Prüfung vor jeder Verarbeitung
        try:
```

```
            validate_captcha(
                token=request.POST.get("eu-captcha-response", ""),
                remote_addr=request.META.get("REMOTE_ADDR"),
            )
        except ValidationError:
            return HttpResponseBadRequest("captcha verification failed")

        # ab hier die bisherige Logik
        email = request.POST["email"]
        message = request.POST["message"]
        return HttpResponse("Danke!")
```

Das Widget legt das Token als Formularfeld ab, deshalb steht es in `request.POST`. Halten Sie Sitekey und Secret in Umgebungsvariablen. Alternativ pflegen Sie die Zugangsdaten im Administrationsbereich von Django unter **Captcha Configurations**. Die als Vorgabe markierte Konfiguration hat Vorrang vor den Werten aus `settings.py`.

Im Trainingsmodus wirft `validate_captcha` keine `ValidationError`, und die Übermittlung läuft durch. Der Trainingsmodus greift bei einem unbekannten Sitekey, bei einem falschen Secret und bei abgeschaltetem Schutz. Prüfen Sie deshalb immer mit den echten Zugangsdaten aus dem Kundenportal.

Siehe **Django**.

5.7.5.6 Integration prüfen

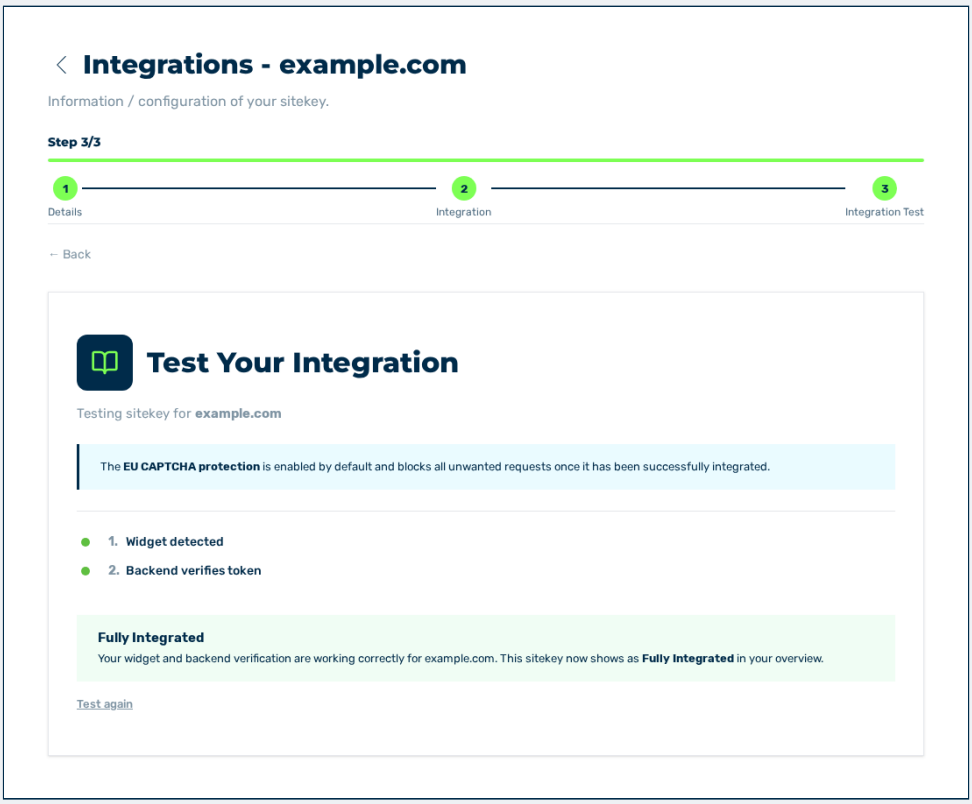


Abbildung 154: Ansicht Integration Test mit dem Ergebnis Fully Integrated

Prüfen Sie zum Schluss in der Ansicht **Integration Test** des Sitekeys, ob Frontend und Backend zusammenspielen. Danach trägt der Sitekey den Status **Fully Integrated**.

Siehe **Integration testen**.

6. FAQ und Fehlerbehebung

6.1 FAQ

Diese Seite beantwortet die Fragen, die vor und während einer Integration am häufigsten auftreten. Jede Antwort verweist auf den Abschnitt mit der ausführlichen Beschreibung.

6.1.1 Muss der Besucher ein Rätsel lösen?

Nein. Das Widget sammelt Signale des Browsers, und der Browser löst eine Rechenaufgabe im Hintergrund. Der Besucher sieht ausschließlich ein kleines Myra-Logo. Es gibt keine Rätsel, keine Bildauswahl und keine Eingabe.

Siehe **Funktionsweise**.

6.1.2 Benötigt die Website ein Cookie-Banner für das Widget?

Nein. Das Myra EU CAPTCHA arbeitet ohne HTTP-Cookies und ohne dauerhaften Browserspeicher.

Siehe **Einführung**.

6.1.3 Wie lange dauert eine Challenge?

Die Dauer richtet sich nach der Schwierigkeitsstufe. Es gibt acht Stufen von 0 bis 7. Stufe 0 ist nach etwa 600 Millisekunden gelöst, Stufe 7 nach etwa 30 Minuten. Legitime menschliche Besucher erhalten fast immer Stufe 0 oder 1.

Siehe **Challenge dauert ungewöhnlich lange**.

6.1.4 Kann ich den Schwierigkeitsgrad selbst festlegen?

Sie legen je Sitekey einen Anfangswert fest. Die Einstellung umfasst die Stufen 0 bis 3. Die höheren Stufen vergibt die Risikoerkennung selbst, sobald verdächtiges Verhalten auftritt.

Siehe **Sitekey konfigurieren**.

6.1.5 Wie oft lässt sich ein Token verwenden?

Jedes Token gilt einmal. Eine zweite Prüfung desselben Tokens antwortet mit der Fehlerkennung `timeout-or-duplicate`. Dieselbe Kennung steht auch für eine abgelaufene Challenge.

Siehe **Client-Token prüfen**.

6.1.6 Warum meldet der Endpunkt einen Erfolg, obwohl keine Prüfung stattfand?

Die Antwort enthält in diesem Fall `success: true` zusammen mit `train: true` . Der Wert `train: true` bedeutet, dass keine echte Prüfung stattgefunden hat. Jede Übermittlung gilt dann als erfolgreich. Der Fall tritt bei einem unbekannten Sitekey, bei einem nicht passenden Secret, bei abgeschaltetem Schutz des Sitekeys und bei jeder weiteren Störung der Prüfung ein.

Werten Sie im Wirkbetrieb immer beide Felder aus und prüfen Sie bei `train: true` sofort die Werte von `sitekey` und `secret` .

Siehe **Client-Token prüfen**.

6.1.7 Was zählt als Assessment?

Ein Assessment zählt, sobald das Widget die Prüfung im Hintergrund beginnt. Die Prüfung im Hintergrund und der serverseitige Aufruf zählen gemeinsam als **ein** Assessment. Der Verifizierungsmodus ändert daran nichts.

Siehe **Funktionsweise**.

6.1.8 Was geschieht bei Überschreitung der monatlichen Obergrenze?

Die monatliche Obergrenze für Assessments wird derzeit nicht durchgesetzt. Ein Überschreiten der Grenze setzt das Widget nicht aus.

Siehe **Reiter Plans**.

6.1.9 Wie viele Sitekeys enthält ein Plan?

Alle Pläne enthalten unbegrenzt viele Sitekeys. Vorgesehen ist ein Sitekey je Domain.

Siehe **Reiter Plans**.

6.1.10 Lässt sich ein Sitekey für mehrere Domains verwenden?

Vorgesehen ist ein Sitekey je Domain. Das System prüft die Eindeutigkeit der Domain nicht, darum entstehen mehrere Sitekeys für dieselbe Domain unbemerkt.

Siehe **Doppelter Sitekey für dieselbe Domain**.

6.1.11 Was geschieht nach dem Ende der Testphase?

Das Widget gibt den Fehlercode `TRIAL_EXPIRED` zurück und verhält sich durchlässig. Besucher werden nicht blockiert, und Ihre Formulare bleiben erreichbar. Der Schutz vor Bots endet. Nach der Buchung eines Plans wird der Schutz innerhalb von 60 Sekunden nach der erfolgreichen Zahlung wieder aktiv.

Siehe **Testphase abgelaufen**.

6.1.12 Sind meine Formulare gesperrt, wenn die API ausfällt?

Nein. Das Widget verhält sich bei einer Störung durchlässig. Die Formulare bleiben für Besucher erreichbar, der Schutz wirkt in dieser Zeit jedoch nicht.

Siehe **API nicht erreichbar**.

6.1.13 Wo finde ich Sitekey und Secret?

Beide Werte stehen in der Ansicht **Details** des Sitekeys.

Siehe **Ansicht Details**.

6.1.14 Darf das Secret im Quelltext der Seite stehen?

Nein. Das Secret bleibt auf Ihrem Server. Rufen Sie den Endpunkt `/verify` ausschließlich von Ihrem Server auf. Im Quelltext der Seite steht allein der öffentliche Sitekey.

Siehe **Client-Token prüfen**.

6.1.15 Wie prüfe ich, ob die Integration vollständig ist?

Die Ansicht **Integration Test** prüft beide Teile getrennt: das Laden des Widgets und die serverseitige Prüfung des Tokens. Das Ergebnis **Fully Integrated** bestätigt eine vollständige Integration.

Siehe **Integration testen**.

6.1.16 Welche Adressen muss meine Content Security Policy freigeben?

Die Direktiven `script-src` und `frame-src` müssen die Adresse `https://cdn.eu-captcha.eu` enthalten, die Direktive `connect-src` die Adresse `https://api.eu-captcha.eu` . Wenn eine dieser Freigaben fehlt, dann blockiert der Browser das Widget, obwohl der Dienst erreichbar ist.

Siehe **Content Security Policy**.

6.2 Testphase abgelaufen

Das Widget gibt den Fehlercode `TRIAL_EXPIRED` zurück. Der Schutz wirkt nicht mehr.

6.2.1 Ursache

Die Testphase Ihres Kontos ist abgelaufen, und es ist kein kostenpflichtiger Plan aktiv.

6.2.2 Verhalten

Das Widget verhält sich in diesem Fall durchlässig: Besucher werden nicht blockiert, und Ihre Formulare bleiben erreichbar. Der aktive Schutz vor Bots endet jedoch.

6.2.3 Lösung

- ▶ Buchen Sie einen Plan. Siehe **Plan buchen**.
- ➔ Der Schutz wird innerhalb von 60 Sekunden nach der erfolgreichen Zahlung wieder aktiv.

6.3 Challenge dauert ungewöhnlich lange

Ein Besucher wartet auffällig lange, bevor er das Formular absenden kann.

6.3.1 Ursache

Das Myra EU CAPTCHA kennt keine fehlgeschlagene Challenge. Eine Challenge wird entweder gelöst, oder ihre Lösung dauert länger. Eine lange Laufzeit weist auf Datenverkehr mit hohem Risiko hin oder auf eine falsche Einstufung eines legitimen Besuchers.

6.3.2 Lösung

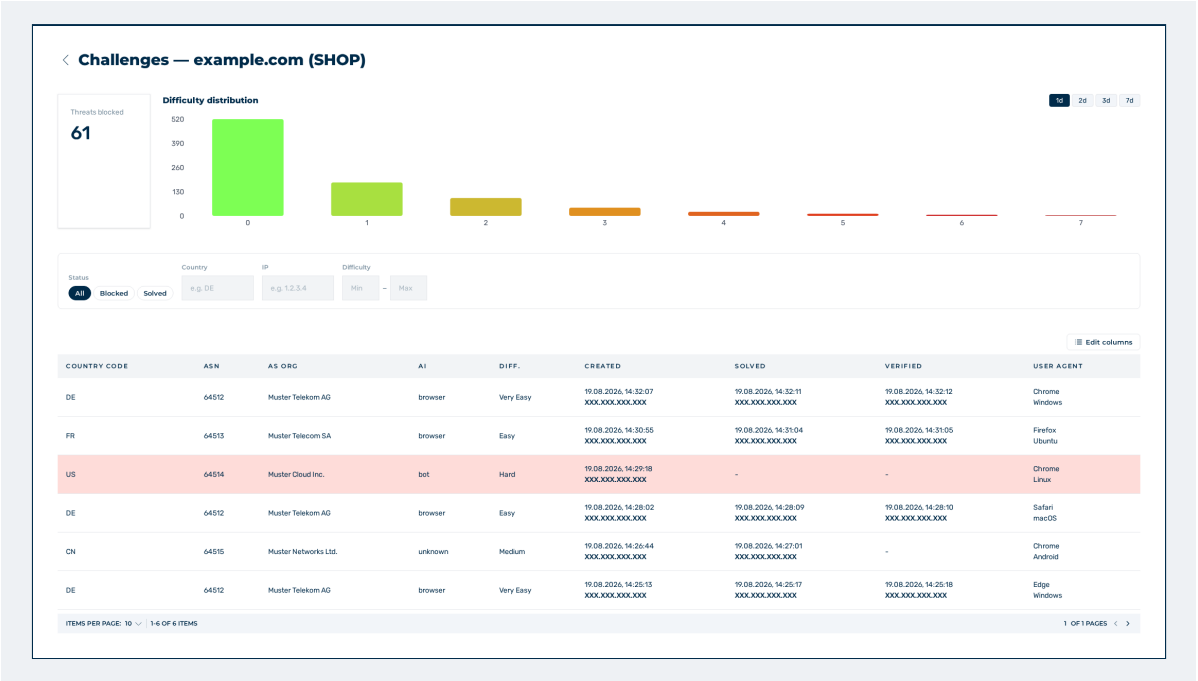


Abbildung 155: Ansicht Challenges

- Prüfen Sie in der Ansicht **Challenges** die Schwierigkeitsgrade der betroffenen Anfragen. Siehe **Ansicht Challenges**.
- Setzen Sie den Wert **Initial difficulty value per challenge 0-3** auf 0. Siehe **Sitekey konfigurieren**.
- Wenn ein legitimer Besucher weiterhin nicht durchkommt, dann wenden Sie sich mit den Angaben zur Sitzung an den Support.

6.4 Grund für eine Blockierung nicht sichtbar

Die Oberfläche und die API nennen nicht, warum eine Anfrage blockiert wurde.

6.4.1 Ursache

Die Gründe für eine Blockierung werden bewusst nicht ausgegeben. Wer die Erkennungslogik kennt, kann sie gezielt umgehen.

6.4.2 Lösung

Abbildung 156: Ansicht Challenges

- Werten Sie in der Ansicht **Challenges** die Merkmale der blockierten Anfragen aus. Beachten Sie besonders die Spalten **ASN**, **Country Code** und **User Agent**. Siehe **Ansicht Challenges**.
- Wenn legitime Besucher regelmäßig blockiert werden, dann wenden Sie sich an den Support.

6.5 Anmeldung über Passkey schlägt fehl

Bei der Anmeldung oder beim Anlegen eines Passkeys meldet das Portal **Invalid attestation object. Unexpected object.**

6.5.1 Ursache

Ein bekannter Fehler in der Verarbeitung des Passkeys. Der Fehler ist erfasst und noch nicht behoben.

6.5.2 Lösung

► Melden Sie sich mit E-Mail-Adresse und Passwort an. Siehe **Anmelden**.

Alternativ stehen die folgenden Verfahren zur Verfügung:

- Anmeldung über Google
- Anmeldung über Apple
- Anmeldung über GitHub
- Anmeldung über Okta
- Anmeldung über SAML und SSO. Siehe **SAML einrichten**.

6.6 Doppelter Sitekey für dieselbe Domain

Für eine Domain bestehen mehrere Sitekeys.

6.6.1 Ursache

Vorgesehen ist ein Sitekey je Domain. Das System prüft die Eindeutigkeit der Domain jedoch nicht. Der Fehler ist erfasst, eine Korrektur ist geplant.

6.6.2 Lösung

Abbildung 157: Bereich Sitekey List

- Öffnen Sie den Bereich **Sitekey List** und suchen Sie über das Suchfeld nach der Domain. Siehe **Bereich Sitekey-Liste**.
- Stellen Sie fest, welcher Sitekey in Ihrer Website eingebunden ist. Der Wert steht im Attribut `data-sitekey` des Widget-Elements.
- Löschen Sie die übrigen Sitekeys. Siehe **Sitekey löschen**.

> Löschen Sie keinen Sitekey, der noch eingebunden ist. Das Widget erhält sonst kein gültiges Token mehr.

6.7 API nicht erreichbar

Die serverseitige Prüfung erreicht den Endpunkt nicht, oder das Widget erhält keine Challenge.

6.7.1 Ursache

Die API des Myra EU CAPTCHA ist vorübergehend nicht erreichbar.

6.7.2 Verhalten

Das Widget verhält sich in diesem Fall durchlässig. Die Formulare bleiben für Besucher erreichbar, der Schutz wirkt jedoch nicht.

6.7.3 Lösung

- Prüfen Sie, ob Ihr Server die Adressen `cdn.eu-captcha.eu` und `api.eu-captcha.eu` erreicht.
- Prüfen Sie die Regeln Ihrer Firewall für ausgehende Verbindungen.
- Wenn die Adressen erreichbar sind und der Fehler bestehen bleibt, dann wenden Sie sich an den Support.

6.8 Widget wird nicht erkannt

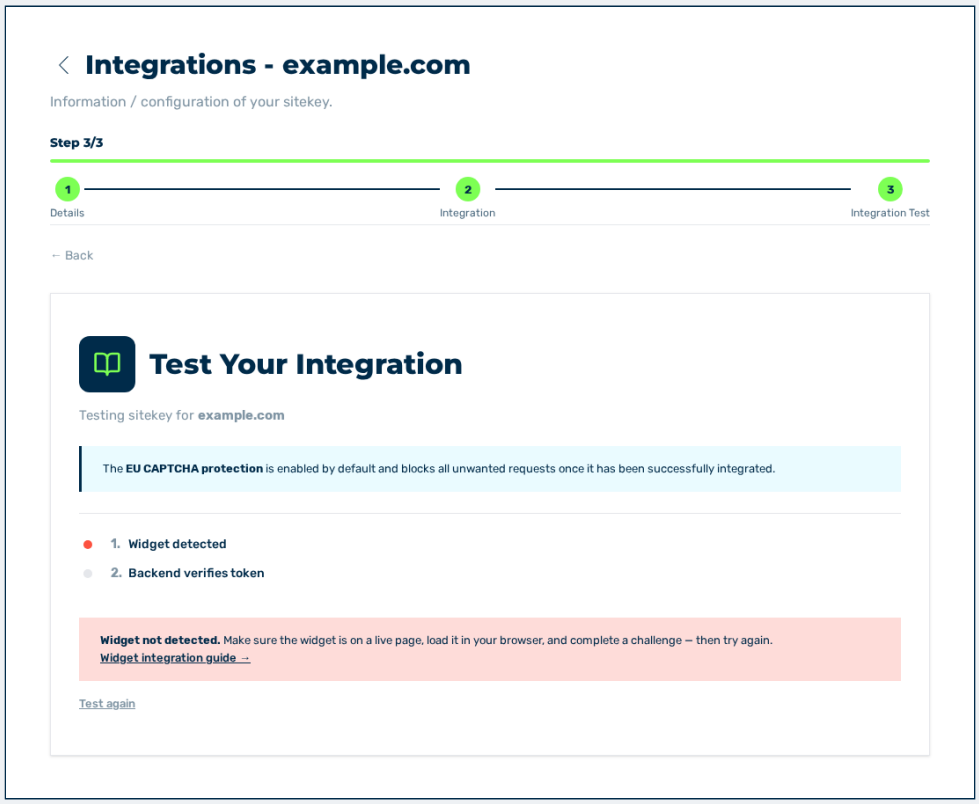


Abbildung 158: Ansicht Integration Test mit der Meldung Widget not detected.

Die Ansicht **Integration Test** meldet **Widget not detected.**, und der Bereich **Sitekey List** zeigt den Status **Not Started** an.

6.8.1 Ursache

Das Widget lädt auf der geschützten Seite nicht, oder es findet sein Element nicht. Die folgenden Ursachen kommen infrage:

- Das Skript-Element oder das Widget-Element fehlt auf der Seite.
- Der Wert im Attribut `data-sitekey` gehört zu einem anderen Sitekey.
- Eine Content Security Policy Ihrer Website blockiert die Adressen des Dienstes. Siehe **Content Security Policy**.

6.8.2 Lösung

- Prüfen Sie, ob das Skript-Element im Bereich `<head>` der Seite steht:

```
<script src="https://cdn.eu-captcha.eu/verify.js" async defer></script>
```

- Prüfen Sie, ob das Widget-Element auf der Seite steht und die Klasse `eu-captcha` trägt:

```
<div class="eu-captcha" data-sitekey="a1b2c3d4-0000-0000-0000-000000000000"></
div>
```

- Prüfen Sie, ob der Wert im Attribut `data-sitekey` dem öffentlichen Schlüssel des Sitekeys entspricht. Siehe **Ansicht Details**.
- Rufen Sie die geschützte Seite auf und prüfen Sie in der Entwicklerkonsole des Browsers, ob die Datei `verify.js` geladen wird.
- Prüfen Sie in der Entwicklerkonsole, ob eine Meldung eine Content Security Policy nennt. Siehe **Content Security Policy**.
- Wiederholen Sie die Prüfung. Siehe **Integration testen**.

Wenn der Test **Widget is working but your server didn't verify the token.** meldet, dann fehlt die serverseitige Prüfung. Siehe **Widget einbinden**.

7. API

7.1 Client-Token prüfen

Rufen Sie den Endpunkt ausschließlich von Ihrem Server auf. Das Secret bleibt auf dem Server und gehört nicht in den Quelltext der Seite.

Senden Sie eine `POST` -Anfrage an den Endpunkt `/verify` unter `https://api.eu-captcha.eu/v1` .

7.1.1 Beschreibung

Der Endpunkt prüft das Proof-of-Work-Token, das `verify.js` im Browser des Besuchers erzeugt hat. Die Prüfung stellt fest, ob die Berechnungen richtig ausgeführt wurden, ob das Token noch nicht verwendet wurde und ob Sitekey und Secret gültig sind.

Das Feld `success` der Antwort nennt das Ergebnis der Prüfung. Werten Sie zusätzlich immer das Feld `train` aus. Siehe die Beschreibung des Felds unter **Antwortfelder**.

Der Endpunkt verlangt keine Autorisierung über eine Kopfzeile. Die Anmeldung erfolgt über das Feld `secret` im Rumpf der Anfrage.

7.1.2 Anfrage

Der Rumpf der Anfrage enthält die folgenden Felder:

Feld	Typ	Erforderlich	Beschreibung
`sitekey`	`string`	ja	Öffentlicher Sitekey der Domain, in der das Widget eingebunden ist. Der Wert steht in der Ansicht **Details** des Sitekeys.
`secret`	`string`	ja	Zum Sitekey gehörendes Secret. Der Wert steht in der Ansicht **Details** des Sitekeys und bleibt auf dem Server.
`client_ip`	`string`	ja	IPv4- oder IPv6-Adresse des Besuchers. Übergeben Sie die Adresse des Besuchers, nicht die eines vorgeschalteten Proxys oder CDN. Werten Sie dazu die Kopfzeilen `X-Forwarded-For` oder `X-Client-IP` aus.
`client_token`	`string`	ja	Token aus `verify.js`. Der Wert erreicht Ihren Server im Formularfeld `eu-captcha-response`. Der Wert ist leer, falls das Widget nicht abgeschlossen hat, etwa bei abgeschaltetem JavaScript.
`client_user_agent`	`string`	ja	Kopfzeile `User-Agent` der Anfrage des Besuchers. Der Wert kennzeichnet die Art des Clients, falls kein Token berechnet wurde.

Senden Sie das Feld `client_token` immer mit, auch mit leerem Wert. Aus unvollständigen und leeren Token lernt der Dienst über alle Prüfungen einschließlich der gescheiterten und erkennt Angriffe dadurch genauer.

Siehe **Ansicht Details**.

7.1.3 Beispiel

Rumpf der Anfrage:

```
{
  "sitekey": "1c87e240-0000-0000-0000-23ac9f99da68",
  "secret": "LqFgQA••••",
  "client_ip": "XXX.XXX.XXX.XXX",
  "client_token": "XVtHUQEx••••",
  "client_user_agent": "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/114.0.0.0 Safari/537.36"
}
```

7.1.4 Antworten

Der Endpunkt liefert die folgenden Statuscodes:

Statuscode	Beschreibung
`200`	Ergebnis der Prüfung. Werten Sie `success` und `train` aus.
`400`	Im Rumpf der Anfrage fehlen erforderliche Felder, oder der Rumpf ist fehlerhaft.
`429`	Die Anzahl der Anfragen ist überschritten. Warten Sie die in der Kopfzeile `Retry-After` genannte Anzahl Sekunden ab und wiederholen Sie die Anfrage.
`500`	Unerwarteter Fehler auf dem Server.

7.1.4.1 Antwortfelder

Die Antwort mit dem Statuscode `200` enthält die folgenden Felder:

Feld	Typ	Beschreibung
`success`	`boolean`	`true`, falls das Token die Prüfung bestanden hat, sonst `false`. Werten Sie immer zusätzlich das Feld `train` aus, da `success` bei `train: true` den Wert `true` erhält.
`train`	`boolean` oder `null`	Nennt, ob eine echte Prüfung stattgefunden hat. Die Werte `false` und `null` stehen für den Regelbetrieb. Der Wert `true` steht für eine übersprungene Prüfung.
`error-codes`	`array`	Nur bei `success: false` vorhanden. Nennt die Gründe des Fehlschlags.

Eine Antwort mit `train: true` bedeutet, dass die Anfrage nicht geprüft wurde und jede Übermittlung als erfolgreich gilt. Der Fall tritt bei einem unbekannten Sitekey, bei einem nicht passenden Secret, bei abgeschaltetem Schutz des Sitekeys und bei jeder weiteren Störung der Prüfung ein. Prüfen Sie im Wirkbetrieb sofort die Werte von `sitekey` und `secret`.

7.1.4.2 Fehlerkennungen

Das Feld `error-codes` enthält die folgenden Werte. Das Format entspricht reCAPTCHA, hCaptcha und Turnstile.

Wert	Bedeutung
`invalid-input-secret`	Das Secret passt nicht zum Sitekey.
`invalid-input-sitekey`	Der Sitekey ist unbekannt.
`invalid-input-response`	Das Feld `client_token` war vorhanden, enthielt aber keinen gültigen Nachweis. Der Wert war fehlerhaft, nicht dekodierbar, ungelöst oder leer.
`timeout-or-duplicate`	Das Token wurde bereits verwendet, oder die Challenge ist abgelaufen. Jedes Token gilt einmal.
`missing-input-secret`	Das Feld `secret` fehlt oder ist keine Zeichenkette.
`missing-input-sitekey`	Das Feld `sitekey` fehlt oder ist keine Zeichenkette.
`missing-input-response`	Das Feld `client_token` fehlt oder ist keine Zeichenkette. Eine leere Zeichenkette gilt als vorhanden und führt zu `invalid-input-response`.
	Das Feld `client_ip` fehlt oder ist keine Zeichenkette.

Wert	Bedeutung
`missing-input-remoteip`	

7.1.4.3 Beispiele der Antwort

Token gültig:

```
{
  "success": true,
  "train": false
}
```

Token ungültig oder bereits verwendet:

```
{
  "success": false,
  "train": false
}
```

Prüfung übersprungen, Zugangsdaten fehlerhaft:

```
{
  "success": true,
  "train": true
}
```

Fehlerantwort mit den Statuscodes `400`, `429` und `500`:

```
{
  "error": "missing_field",
  "message": "Required field 'sitekey' is missing."
}
```

7.1.5 Zugangsdaten prüfen

Zum Prüfen von Sitekey und Secret ohne ein Token siehe **Sitekey und Secret prüfen**.

7.2 Sitekey und Secret prüfen

Rufen Sie den Endpunkt ausschließlich von Ihrem Server auf. Das Secret bleibt auf dem Server und gehört nicht in den Quelltext der Seite.

Senden Sie eine `POST` -Anfrage an den Endpunkt `/verify-credentials` unter `https://api.eu-captcha.eu/v1` .

7.2.1 Beschreibung

Der Endpunkt prüft, ob ein Paar aus Sitekey und Secret gültig und aktiv ist. Ein Token des Besuchers ist dafür nicht erforderlich. Setzen Sie den Aufruf beim Start Ihrer Anwendung oder beim Einrichten ein, um fehlerhafte Zugangsdaten früh zu erkennen.

Im Unterschied zu `/verify` verbraucht der Endpunkt kein Token und führt keine Proof-of-Work-Prüfung aus. Er stellt ausschließlich fest, ob die übergebenen Zugangsdaten bekannt und freigeschaltet sind.

Der Endpunkt steht ab Version 1.1 der API zur Verfügung.

Siehe **Client-Token prüfen**.

7.2.2 Anfrage

Der Rumpf der Anfrage enthält die folgenden Felder:

Feld	Typ	Erforderlich	Beschreibung
`sitekey`	`string`	ja	Öffentlicher Sitekey der Domain, in der das Widget eingebunden ist. Der Wert steht in der Ansicht **Details** des Sitekeys.
`secret`	`string`	ja	Zum Sitekey gehörendes Secret. Der Wert steht in der Ansicht **Details** des Sitekeys und bleibt auf dem Server.

Siehe **Ansicht Details**.

7.2.3 Beispiel

Rumpf der Anfrage:

```
{
  "sitekey": "1c87e240-0000-0000-0000-23ac9f99da68",
  "secret": "LqFgQA••••"
}
```

7.2.4 Antworten

Der Endpunkt liefert die folgenden Statuscodes:

Statuscode	Beschreibung
`200`	Ergebnis der Prüfung der Zugangsdaten.
`400`	Im Rumpf der Anfrage fehlen erforderliche Felder, oder der Rumpf ist fehlerhaft.
`429`	Die Anzahl der Anfragen ist überschritten. Warten Sie die in der Kopfzeile `Retry-After` genannte Anzahl Sekunden ab und wiederholen Sie die Anfrage.
`500`	Unerwarteter Fehler auf dem Server.

7.2.4.1 Antwortfelder

Die Antwort mit dem Statuscode `200` enthält das folgende Feld:

Feld	Typ	Beschreibung
`valid`	`boolean`	`true`, falls der Sitekey vorhanden ist und das Secret dazu passt, sonst `false`.

7.2.4.2 Beispiele der Antwort

Zugangsdaten gültig:

```
{
  "valid": true
}
```

Sitekey unbekannt oder Secret nicht passend:

```
{
  "valid": false
}
```

Fehlerantwort mit den Statuscodes `400`, `429` und `500`:

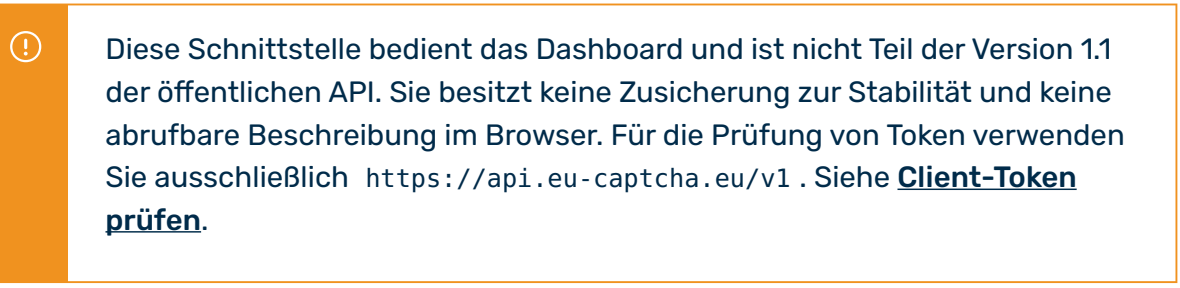
```
{
  "error": "missing_field",
  "message": "Required field 'sitekey' is missing."
}
```

7.3 Sitekeys verwalten

Neben der Verifikations-API betreibt das Myra EU CAPTCHA eine zweite Schnittstelle: die API des Dashboards. Über sie legen Sie Sitekeys an, lesen deren Einstellungen und ändern sie. Das Dashboard selbst verwendet ausschließlich diese Schnittstelle.

Die Basis-URL lautet:

```
https://api-app.eu-captcha.eu/myra-auto-app-api
```



Diese Schnittstelle bedient das Dashboard und ist nicht Teil der Version 1.1 der öffentlichen API. Sie besitzt keine Zusicherung zur Stabilität und keine abrufbare Beschreibung im Browser. Für die Prüfung von Token verwenden Sie ausschließlich `https://api.eu-captcha.eu/v1` . Siehe **Client-Token prüfen**.

7.3.1 Anmelden

Alle Aufrufe benötigen einen Token nach JSON Web Token. Sie erhalten ihn über den Endpunkt `/login` :

```
curl -X POST https://api-app.eu-captcha.eu/myra-auto-app-api/login \
  -H "Content-Type: application/json" \
  -d '{
        "email": "max.mustermann@example.com",
        "password": "••••••••"
      }'
```

Die Antwort enthält den Token:

```
{
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9••••"
}
```

Ist für das Konto die Zwei-Faktor-Authentifizierung eingeschaltet, antwortet der Endpunkt stattdessen mit dem Feld `totp_required` und einem Wert `preAuthToken` . Der Token folgt erst nach der Bestätigung des Einmalkennworts. Siehe **Zwei-Faktor-Authentifizierung einrichten**.

Übergeben Sie den Token bei jedem weiteren Aufruf in der Kopfzeile:

```
Authorization: Bearer <Token>
```

7.3.2 Sitekeys auflisten

```
GET /captcha_sitekeys
```

Der Endpunkt liefert die Sitekeys, die Ihrem Konto gehören oder für die Sie eine Freigabe besitzen. Die folgenden Parameter stehen zur Verfügung:

Parameter	Beschreibung
`page`	Seite der Ergebnisliste.
`itemsPerPage`	Anzahl der Einträge je Seite.
`light`	Mit `light=1` entfallen die aufwendigen Kennzahlen zur Nutzung. Die Antwort enthält dann nur die Stammdaten.

```
curl "https://api-app.eu-captcha.eu/myra-auto-app-api/captcha_sitekeys?
page=1&itemsPerPage=10" \
  -H "Authorization: Bearer <Token>"
```

Die Antwort enthält die Einträge im Feld `member` und die Gesamtzahl im Feld `totalItems`.

7.3.3 Einzelnen Sitekey lesen

```
GET /captcha_sitekeys/{id}
```

`{id}` ist die interne Nummer des Sitekeys, nicht dessen UUID.

7.3.4 Sitekey anlegen

```
POST /captcha_sitekeys
```

Die Anfrage verwendet den Inhaltstyp `application/ld+json` :

```
curl -X POST https://api-app.eu-captcha.eu/myra-auto-app-api/captcha_sitekeys
\
  -H "Authorization: Bearer <Token>" \
  -H "Content-Type: application/ld+json" \
  -d '{
        "domain": "example.com",
        "label": "Kontaktformular"
      }'
```

Den Sitekey und das Secret erzeugt der Server. Der Sitekey ist eine UUID, das Secret ein Zufallswert mit 32 Byte in Base64. Beide Werte stehen in der Antwort.

7.3.5 Sitekey ändern

```
PATCH /captcha_sitekeys/{id}
```

Die Anfrage verwendet den Inhaltstyp `application/merge-patch+json` und enthält nur die Felder, die sich ändern:

```
curl -X PATCH https://api-app.eu-captcha.eu/myra-auto-app-api/
captcha_sitekeys/42 \
  -H "Authorization: Bearer <Token>" \
  -H "Content-Type: application/merge-patch+json" \
  -d '{ "initialDifficulty": 2 }'
```

Ein Sitekey wird nicht entfernt, sondern über das Feld `deleted` stillgelegt:

```
{ "deleted": true }
```

Siehe **Sitekey löschen**.

7.3.6 Felder

Die folgenden Felder setzen Sie beim Anlegen und beim Ändern:

Feld	Anlegen	Ändern	Wertebereich
`domain`	ja	nein	Gültiger Hostname. Die Domain eines bestehenden Sitekeys ändern Sie nicht.
`label`	ja	ja	Höchstens 255 Zeichen.
`train`	ja	ja	`true` oder `false`. Im Trainingsmodus antwortet die Verifikation stets erfolgreich.
`maxParallel`	ja	ja	1 bis 10.
`initialDelay`	ja	ja	3 bis 30 Sekunden.
`initialDifficulty`	ja	ja	0 bis 3.
`wizardCategory`	ja	nein	Kategorie aus dem Installationsassistenten, höchstens 64 Zeichen.
`wizardTechnology`	ja	nein	Technologie aus dem Installationsassistenten, höchstens 64 Zeichen.
`deleted`	nein	ja	`true` legt den Sitekey still.

Liegt ein Wert außerhalb des Bereichs, antwortet der Server mit einem Fehler und nennt das betroffene Feld. Die Bedeutung der Einstellungen beschreibt der Abschnitt **Sitekey konfigurieren**.

7.3.7 Zusätzliche Felder der Antwort

Neben den gespeicherten Werten enthält die Antwort berechnete Felder:

Feld	Bedeutung
`permissionLevel`	Ihr Zugriff auf den Sitekey: `owner`, `write`, `read` oder `none`.
`challengesServed`	Anzahl der ausgelieferten Challenges im jüngsten Zeitraum.
`installationStatus`	`not_started`, `widget_installed` oder `fully_integrated`. Der Wert schreitet nur vorwärts.
`recentActivity`	`active` oder `inactive`.

Die drei letzten Felder entfallen bei einem Aufruf mit `light=1`.

7.3.8 Berechtigungen

Der Zugriff richtet sich nach der Freigabe des Sitekeys:

Freigabe	Lesen	Ändern	Secret in der Antwort
Eigentümer	ja	ja	ja
`write`	ja	ja	ja
`read`	ja	nein	nein, das Feld ist leer

Freigaben verwalten Sie im Dashboard. Siehe **Zugriff auf einen Sitekey gewähren**.

7.4 Statistiken abrufen

Die Zahlen, die das Dashboard in den Ansichten **Dashboard** und **Challenges** darstellt, stammen aus der API des Dashboards. Über dieselben Endpunkte rufen Sie die Werte für eigene Auswertungen ab.

Die Basis-URL lautet:

```
https://api-app.eu-captcha.eu/myra-auto-app-api
```

Alle Aufrufe verwenden die Methode `GET` und benötigen einen Token nach JSON Web Token in der Kopfzeile `Authorization`. Siehe **Sitekeys verwalten**.

Diese Schnittstelle bedient das Dashboard und ist nicht Teil der Version 1.1 der öffentlichen API. Sie besitzt keine Zusicherung zur Stabilität und keine abrufbare Beschreibung im Browser.

7.4.1 Endpunkte

Die folgenden Endpunkte stehen zur Verfügung:

Endpunkt	Inhalt
`/captcha-stats-user`	Aktionen über alle Sitekeys, auf die Sie Zugriff haben.
`/captcha-stats-sitekey`	Aktionen eines einzelnen Sitekeys.
`/captcha-challenges-per-user`	Anzahl der Challenges über alle zugänglichen Sitekeys.
`/recent-captcha-challenges-for-sitekey`	Die jüngsten Challenges eines Sitekeys als Liste.
`/captcha-difficulty-distribution`	Verteilung der Schwierigkeitsgrade eines Sitekeys.
`/captcha-threats-blocked`	Anzahl der abgewehrten Zugriffe eines Sitekeys.

Ein Sitekey, auf den Sie keinen Zugriff haben, liefert die Antwort `{"success": false}`. Dasselbe gilt, wenn der Parameter `sitekey` fehlt, sofern der Endpunkt ihn benötigt.

7.4.2 Zeitraum wählen

Die Endpunkte zu den Aktionen und den Challenges verwenden dieselben Parameter für den Zeitraum:

Parameter	Vorgabe	Beschreibung
`sitekey`	leer	UUID des Sitekeys. Bleibt der Wert leer, gelten alle zugänglichen Sitekeys.
`max_days`	`5`	Länge des Zeitraums in Tagen.
`start_date`	leer	Erster Tag des Zeitraums im Format `JJJJ-MM-TT`. Ohne Angabe endet der Zeitraum am heutigen Tag.
`per_day`	`0`	Mit `per_day=1` liefert `/captcha-stats-user` die Werte je Tag statt als Summe.

```
curl "https://api-app.eu-captcha.eu/myra-auto-app-api/captcha-stats-user?
max_days=30&per_day=1" \
  -H "Authorization: Bearer <Token>"
```

Die Antwort enthält je Zeile die Art der Aktion, die Anzahl und — bei `per_day=1` — das Datum:

```
[
  { "action_type": "ACC", "date_at": "2026-08-01", "count": 1284 },
  { "action_type": "BIT", "date_at": "2026-08-01", "count": 37 }
]
```

7.4.3 Arten von Aktionen

Das Feld `action_type` unterscheidet die folgenden Werte:

Wert	Bedeutung
`VFC`	Eine Challenge wurde ausgeliefert.
`ASC`	Eine Challenge wurde gelöst.
`VSC`	Der Client hat die Challenge ohne Erfolg gemeldet. Der Wert gilt als neutral, nicht als Blockierung.
`ACC`	Die Prüfung des Tokens war erfolgreich, der Zugriff ist zugelassen.
`BCI`	Der Nachweis zur Challenge war falsch.
`BIC`	Der Client hat eine andere Challenge eingereicht, als ihm zugewiesen war.
`BCR`	Der Token war bereits verwendet oder abgelaufen.
`BIT`	Die Anfrage enthielt keinen auswertbaren Token.

Alle Werte, die mit `B` beginnen, sind blockierte Zugriffe.

Das Dashboard fasst die Werte wie folgt zusammen:

Kennzahl	Enthaltene Aktionen
Verified	alle Aktionen `ACC`
Blocked	alle Aktionen, deren Wert mit `B` beginnt
Challenges solved	die Aktionen `ASC` und `VSC`
Block rate	Anteil der blockierten Zugriffe an der Summe aus blockierten und zugelassenen Zugriffen, in Prozent

Siehe **Dashboard**.

7.4.4 Jüngste Challenges

```
GET /recent-captcha-challenges-for-sitekey
```

Der Endpunkt verwendet die folgenden Parameter:

Parameter	Vorgabe	Beschreibung
`sitekey`	—	UUID des Sitekeys. Der Parameter ist erforderlich.
`max_results`	`30`	Anzahl der Einträge je Seite.
`page`	`0`	Seite der Ergebnisliste.

Die Antwort enthält die Einträge im Feld `member`, die Gesamtzahl im Feld `totalItems` sowie die Domain und die Bezeichnung des Sitekeys.

Das Feld `created_ai_result` nennt die Einschätzung der Anfrage. Es besitzt die Werte `browser`, `bot` und `unknown`. Siehe **Challenges**.

7.4.5 Verteilung der Schwierigkeit

```
GET /captcha-difficulty-distribution
```

Der Endpunkt verwendet die folgenden Parameter:

Parameter	Vorgabe	Beschreibung
`sitekey`	—	UUID des Sitekeys. Der Parameter ist erforderlich.
`days`	`1`	Zeitraum in Tagen, 1 bis 7. Werte darüber gelten als 7.

Die Antwort nennt je Schwierigkeitsgrad die Anzahl der Challenges. Die Grade der Anzeige reichen weiter als der Startwert eines Sitekeys, denn das Myra EU CAPTCHA erhöht die Schwierigkeit bei Verdacht selbsttätig. Siehe **Sitekey konfigurieren**.

7.4.6 Abgewehrte Zugriffe

```
GET /captcha-threats-blocked
```

Der Endpunkt verwendet dieselben Parameter wie die Verteilung der Schwierigkeit und antwortet mit einer einzelnen Zahl:

```
{ "count": 214 }
```

8. Referenz

8.1 Glossar

Diese Seite erklärt die Begriffe, die in dieser Dokumentation vorkommen.

8.1.1 Begriffe

Die folgenden Begriffe sind gebräuchlich:

Begriff	Bedeutung
Assessment	Eine geprüfte Anfrage. Die Prüfung im Hintergrund und die zugehörige serverseitige Verifikation zählen zusammen als ein Assessment. Die Anzahl der Assessments je Monat bestimmt den Plan. Siehe **Plan buchen**.
Bouncer	Eigenständiger vorgelagerter Server, der die Entscheidungen von CrowdSec mit dem Myra EU CAPTCHA verbindet. Siehe **CrowdSec**.
Challenge	Rechenaufgabe, die der Browser des Besuchers im Hintergrund löst. Aus der Lösung entsteht der Token. Siehe **Funktionsweise**.
Content Security Policy	Richtlinie einer Website, die dem Browser vorgibt, aus welchen Quellen er Inhalte laden darf. Sie muss die Adressen des Myra EU CAPTCHA freigeben. Siehe **Content Security Policy**.
Org Admin	Rolle innerhalb einer Organisation. Sie berechtigt zur Verwaltung der Mitglieder und der Anmeldung über SAML. Siehe **Rolle Org Admin vergeben und entziehen**.
Passkey	Schlüssel auf dem Gerät der Nutzenden, der das Passwort bei der Anmeldung ersetzt. Siehe **Passkey erstellen**.
Proof-of-Work	Verfahren, bei dem der Client eine Rechenaufgabe löst, bevor eine Anfrage angenommen wird. Für einzelne Nutzende ist der Aufwand unmerklich, für den massenhaften Zugriff durch Bots wird er teuer. Siehe **Funktionsweise**.
SAML	Verfahren für die Anmeldung über den Identity Provider eines Unternehmens. Siehe **SAML einrichten**.
Schwierigkeitsgrad	Aufwand einer Challenge. Die Auswertung stellt die Grade 0 bis 7 dar. Der Startwert je Sitekey umfasst die Grade 0 bis 3. Siehe **Sitekey konfigurieren**.
Secret	Geheimer Wert eines Sitekeys. Er gehört ausschließlich auf den Server und weist die Aufrufe der API aus. Siehe **Details**.
Sitekey	

Begriff	Bedeutung
	Öffentlicher Wert, der eine geschützte Domain kennzeichnet. Er steht im Widget der Seite. Siehe **Sitekey anlegen**.
Token	Ergebnis einer gelösten Challenge. Das Widget legt ihn im Formularfeld `eu-captcha-response` ab. Der Server prüft ihn über den Endpunkt `/verify`. Jeder Token gilt nur einmal. Siehe **Client-Token prüfen**.
train-Modus	Zustand, in dem die API die Prüfung überspringt und `success` fest auf `true` setzt. Die Antwort enthält dann `train: true`. Siehe **Sitekey konfigurieren**.
verify.js	Skript des Widgets. Es erzeugt die Challenge im Browser und legt den Token im Formular ab. Siehe **Widget einbinden**.
Widget	Baustein, den Sie in Ihr Formular einbinden. Er führt die Challenge aus und übergibt den Token. Siehe **Widget einbinden**.

8.2 Content Security Policy

Eine Content Security Policy (CSP) ist eine Richtlinie, mit der eine Website festlegt, aus welchen Quellen der Browser Inhalte laden darf. Wenn Ihre Website eine solche Richtlinie sendet, dann lädt der Browser das Widget des Myra EU CAPTCHA nur, wenn die Richtlinie die Adressen des Dienstes ausdrücklich erlaubt.

8.2.1 Erforderliche Freigaben

Das Widget verwendet zwei Adressen. Die folgende Tabelle nennt die Direktiven, die diese Adressen enthalten müssen:

Direktive	Adresse	Wofür
`script-src`	`https://cdn.eu-captcha.eu`	Lädt die Datei `verify.js` in die geschützte Seite.
`frame-src`	`https://cdn.eu-captcha.eu`	Lädt die Datei `check.html` in das versteckte Iframe, in dem die Prüfung abläuft.
`connect-src`	`https://api.eu-captcha.eu`	Fragt vor der Prüfung einen Wert für das Private-Access-Token-Verfahren ab. Dieses Verfahren unterstützt derzeit nur Safari.

`frame-src` ist genauso notwendig wie `script-src`. Die Datei `verify.js` erzeugt das Iframe erst zur Laufzeit, deshalb fällt eine fehlende Freigabe nicht beim Einbinden auf, sondern erst im Betrieb.

8.2.2 Beispiel einer Richtlinie

Der folgende Header erlaubt dem Browser alles, was das Widget benötigt:

```
Content-Security-Policy: script-src 'self' https://cdn.eu-captcha.eu; frame-
src 'self' https://cdn.eu-captcha.eu; connect-src 'self' https://api.eu-
captcha.eu
```

Die Direktive `default-src` gilt als Rückfallebene für alle Direktiven, die Ihre Richtlinie nicht einzeln aufführt. Wenn Ihre Richtlinie nur `default-src` enthält, dann nehmen Sie die beiden Adressen dort auf.

8.2.3 Wirkung einer fehlenden Freigabe

Eine fehlende Freigabe wirkt sich je nach Direktive verschieden aus:

Fehlende Direktive	Wirkung
`script-src`	Der Browser lädt `verify.js` nicht. Auf der Seite erscheint kein Widget, und das Formular enthält kein Feld `eu-captcha-response` .
`frame-src`	Die Datei `verify.js` lädt, das Iframe bleibt jedoch leer. Das Widget erzeugt keinen Token.
`connect-src`	Das Widget arbeitet weiter. In Safari entfällt das Private-Access-Token-Verfahren.

Der Browser blockiert die Datei, bevor er die Anfrage sendet. Im Netzwerkbereich der Entwicklerwerkzeuge steht die Anfrage deshalb ohne Statuscode. Dieser Befund sieht wie eine Störung des CDN aus, obwohl der Dienst erreichbar ist und antwortet. Siehe **API nicht erreichbar**.

8.2.4 Darstellung des Widgets

Das Widget setzt seine Größe und die Position des Myra-Logos über Stil-Attribute direkt an den erzeugten Elementen. Wenn Ihre Richtlinie eine Direktive `style-src` oder `style-src-attr` ohne den Wert `'unsafe-inline'` enthält, dann erscheint das Widget an falscher Stelle oder in falscher Größe. Die Prüfung selbst läuft weiter.

8.2.5 Richtlinie prüfen

Um die Richtlinie Ihrer Website zu prüfen, gehen Sie wie folgt vor:

- ▶ Rufen Sie die geschützte Seite in einem Browser auf.
- ▶ Öffnen Sie die Entwicklerwerkzeuge des Browsers.
- ▶ Prüfen Sie in der Konsole, ob eine Meldung eine der drei Adressen nennt.
- ▶ Prüfen Sie in der Netzwerkanalyse den Antwortheader `Content-Security-Policy` der Seite.

➔ Die Meldungen der Konsole nennen die Direktive, die den Zugriff verweigert hat.

Eine Richtlinie steht entweder im Antwortheader `Content-Security-Policy` oder in einem Element `<meta http-equiv="Content-Security-Policy">` im Bereich `<head>` der Seite. Beide Formen wirken zusammen: Der Browser wendet jede Richtlinie einzeln an, und die Freigabe muss deshalb in jeder von ihnen stehen.

Siehe **Widget wird nicht erkannt** und **Funktionsweise**.