



Wie man den Gmail SMTP Server benutzt, um kostenlos E-Mails zu versenden

Die meisten Leute kennen Gmail für sein sauberes Interface und seine nützlichen Funktionen, wie [Suchmaschinen](#) und [Add-Ons](#). Aber du kannst Gmail auch für mehr nutzen: den Gmail SMTP-Server.

Mit dem Gmail SMTP-Server kannst du von deinem Gmail-Konto aus E-Mails mit anderen E-Mail-Clients wie Outlook oder Thunderbird versenden. **Aber noch wichtiger ist, dass du den SMTP-Server von Gmail auch nutzen kannst, um E-Mails von deiner WordPress-Seite aus zu versenden.** Kostenlos!

Das ist eine wirklich großartige Möglichkeit, [die E-Mails deiner WordPress-Seite zuverlässiger zu machen](#), ohne dass du Geld für einen speziellen E-Mail-Versandservice ausgeben musst. Mit Gmail kannst du **bis zu 500 E-Mails pro Tag** versenden, was für die meisten WordPress-Seiten mehr als genug ist.

In diesem Beitrag decken wir alles ab, was du über den Gmail SMTP-Server wissen musst, einschließlich

Sieh dir unsere [Videoanleitung](#) zur Verwendung des Gmail SMTP-Servers an, um E-Mails kostenlos zu versenden

Wie man den SMTP-Server für Gmail findet

Fangen wir mit der wichtigsten Frage an – was ist der Gmail-SMTP-Server?

Um den Gmail SMTP Server zu finden, kannst du diese Angaben verwenden:

- **Gmail SMTP-Serveradresse:** smtp.gmail.com
- **Gmail SMTP Name:** Dein voller Name
- **Gmail SMTP-Benutzername:** Deine vollständige Gmail-Adresse (z.B. you@gmail.com)
- **Gmail SMTP Passwort:** Das Passwort, mit dem du dich bei Gmail einloggst
- **Gmail SMTP-Port (TLS):** 587
- **Gmail SMTP-Port (SSL):** 465

Es gibt auch einige andere gebräuchliche (aber nicht universelle) Gmail-SMTP-Einstellungen, die du finden kannst, wenn du versuchst, Dinge einzurichten. Hier ist, wie du reagierst, wenn du ihnen begegnest:

- **Benötigt SSL:** Ja
- **Benötigt TLS:** Ja
- **Erfordert Authentifizierung / Authentifizierung verwenden:** Ja
- **Benötigt sichere Verbindung / Sichere Verbindung verwenden:** Ja

Gmail SMTP Server FAQs

Bevor du eintauchst, lass uns einige allgemeine Fragen zum Gmail SMTP-Server beantworten.

Was ist die Adresse des Gmail SMTP-Servers?

Wie wir oben beschrieben haben, ist die standardmäßige Serveradresse **smtp.gmail.com**. Du kannst dich mit deiner vollständigen Gmail-E-Mail-Adresse und deinem Google-Passwort einloggen.

Kann ich den Gmail SMTP Server für das Versenden von E-Mails benutzen?

Ja, das kannst du. Wenn du andere E-Mail-Clients wie Thunderbird oder Outlook verwendest, kannst du die Details des Gmail SMTP-Servers verwenden, um trotzdem E-Mails über dein Gmail-Konto zu versenden.

Denke aber daran, dass SMTP nur für den E-Mail-Versand gedacht ist. Wenn du auch E-Mails an dein Gmail-Konto in einem anderen E-Mail-Programm empfangen möchtest, musst du POP3 oder IMAP verwenden. Du kannst diese Einstellungen finden, indem du deine Gmail-Einstellungen öffnest und zum Reiter **Forwarding and POP/IMAP** gehst.

Kann ich den Gmail SMTP Server benutzen, um WordPress Transaktions-E-mails zu versenden?

Auch ja! WordPress sendet eine Menge einfacher [Transaktions-E-Mails](#) für Dinge wie Passwortrücksetzung, Benachrichtigungen usw. und du kannst all diese E-Mails über den Gmail SMTP-Server zustellen.

Mit nur einem kostenlosen Gmail-Konto kannst du bis zu 500 E-Mails pro Tag verschicken, was deutlich höher ist als die Limits, die [andere kostenlose SMTP-Server](#) haben.

Wenn du ein [kostenpflichtiges Google Workspace-Konto \(früher G Suite\)](#) hast, erhöht Google dein Limit noch weiter und lässt dich bis zu 2.000 E-Mails pro Tag versenden. Du wirst auch in der Lage sein, E-Mails von deinem eigenen [benutzerdefinierten Domainnamen](#) anstelle deiner Gmail-Adresse zu versenden. Das heißt, du kannst von **you@yoursite.com** statt von **you@gmail.com** aus senden. Dazu musst du außerdem [Google Workspace MX-Einträge einrichten](#), um dein Google Workspace-Konto [mit deinem benutzerdefinierten Domainnamen](#) zu verbinden.

Hinweis: Technisch gesehen sind die Sendelimits nicht „pro Tag“. Stattdessen gelten die Limits für einen „rollenden 24-Stunden-Zeitraum“. Zum Beispiel könntest du am Montag nicht 500 E-Mails um 23:59 Uhr versenden und dann weitere 500 E-Mails um 12:01 Uhr am Dienstag.

Funktioniert der Gmail-SMTP-Server noch mit Zwei-Faktor-Authentifizierung?

Ja! Du kannst den SMTP-Server benutzen, auch wenn du die Zwei-Faktor-Authentifizierung in deinem Google-Konto aktiviert hast. Allerdings musst du ein App-Passwort generieren, damit die App weiterhin eine Verbindung herstellen kann.

Du kannst ein App-Passwort generieren, indem du [diese Seite besuchst](#), während du in deinem Google-Konto eingeloggt bist.

Wenn du jedoch den SMTP-Server von Google Mail benutzen möchtest, um die E-Mails deiner WordPress-Seite zu versenden, empfehlen wir diese Methode nicht zu verwenden.

Stattdessen solltest du die Gmail API Methode verwenden, die wir im Tutorial weiter unten beschreiben werden. So kannst du E-Mails über die API von Gmail versenden, anstatt nur die SMTP-Serverdetails einzugeben, was auch den Vorteil hat, Probleme mit der Zwei-Faktor-Authentifizierung zu vermeiden.

Wie man WordPress so konfiguriert, dass E-Mails über den Gmail SMTP-Server gesendet werden

Nun lass uns darüber sprechen, wie du den Gmail SMTP-Server nutzen kannst, um die Transaktions-E-Mails deiner WordPress-Seite **kostenlos** zu versenden. Diese Methode funktioniert großartig und wird die Zuverlässigkeit der E-Mails deiner Seite gegenüber der standardmäßigen [PHP-Mail-Methode](#) verbessern.

Um dies einzurichten, musst du eine Google-App erstellen, damit du dich über die API mit dem Gmail SMTP-Server verbinden kannst, anstatt nur die SMTP-Details einzugeben. Dazu sind eine ganze Reihe von Schritten nötig. Dies ist nur ein einmaliger Einrichtungsprozess. Das heißt, sobald du dir die 30-60 Minuten Zeit nimmst, um dieses Setup durchzuführen, wird deine Seite weiterhin auf Autopilot funktionieren.

Zusätzlich zum Erstellen einer Google-App benötigst du auch die Hilfe eines WordPress SMTP-Plugins. Wir werden das [kostenlose Post SMTP Mailer/Email Log Plugin](#) verwenden, aber das kostenlose [WP Mail SMTP Plugin](#) ist auch eine gute Option, die die Gmail API unterstützt.

Der grundlegende Prozess läuft wie folgt ab:

1. Installiere das Post SMTP Mailer/Email Log Plugin (du musst dies zuerst tun, um die URLs zu erhalten, die du in deiner Google Anwendung verwenden kannst).
2. Erstelle eine Google-App. Dies ist der komplizierteste Teil des Prozesses, aber wir werden dich durch jeden Schritt führen.
3. Füge deine Google App API-Schlüssel zu den Post SMTP Mailer/Email Log-Einstellungen hinzu.
4. Sende eine Test-E-Mail, um sicherzustellen, dass alles funktioniert.

Hinweis – dieses Tutorial konzentriert sich auf die Verwendung eines kostenlosen Gmail-Accounts. Du kannst jedoch einem ähnlichen Prozess folgen, um deine Seite so zu konfigurieren, dass sie E-Mails über dein Google Workspace (G Suite) Konto versendet.

1. Installiere und konfiguriere den Post SMTP Mailer/Email Log

Um anzufangen, musst du das [kostenlose Post SMTP Mailer/Email Log Plugin von WordPress.org installieren und aktivieren](#). Damit kannst du deine WordPress-Seite so konfigurieren, dass sie E-Mails über den Gmail API/SMTP-Server versendet.

Nachdem du das Plugin aktiviert hast, gehe auf den **Post SMTP** Reiter in deinem WordPress Dashboard und klicke auf den Link **Show All Settings** unter dem großen **Start the Wizard** Button.

Gehe dann zum Reiter **Message** und stelle deine „von“-E-Mail-Adresse und deinen Namen ein. Du kannst deine Gmail-Adresse als Absenderadresse verwenden oder eine andere E-Mail-Adresse, wenn dir das lieber ist.

Als nächstes gehe zurück zum **Account** Reiter und wähle **Gmail API** im **Type** Drop-Down Menü. Sobald du diese Wahl getroffen hast, solltest du einige zusätzliche Optionen in dem **Authentication** – Kästchen unten sehen. Lass diese Seite geöffnet, denn im nächsten Schritt benötigst du die Angaben zu **Authorized JavaScript** und **Authorized redirect URI**:

The screenshot shows the 'Transport' and 'Authentication' settings for a WordPress site. The 'Type' dropdown is set to 'Gmail API'. The 'Mailer Type' is 'PostSMTP'. The 'Authentication' section shows 'Authorized JavaScript origins' and 'Authorized redirect URI' both set to their respective URLs. A 'Save Changes' button is at the bottom.

Transport

Choose SMTP or a vendor-specific API:

Type: Gmail API

Mailer Type: PostSMTP

*Beta Feature: Change this to **PHPMailer** only if you see `wp_mail` conflict message, conflicts when another plugin uses the same function.*

Authentication

Attention! Check this article how to configure Gmail/Gsuite OAuth: [Read Here](#)

Authorized JavaScript origins: `https://test.kinsta.cloud`

Authorized redirect URI: `https://test.kinsta.cloud/wp-admin/options-general.php?page=postman`

Client ID: Required

Client Secret: Required

[Save Changes](#)

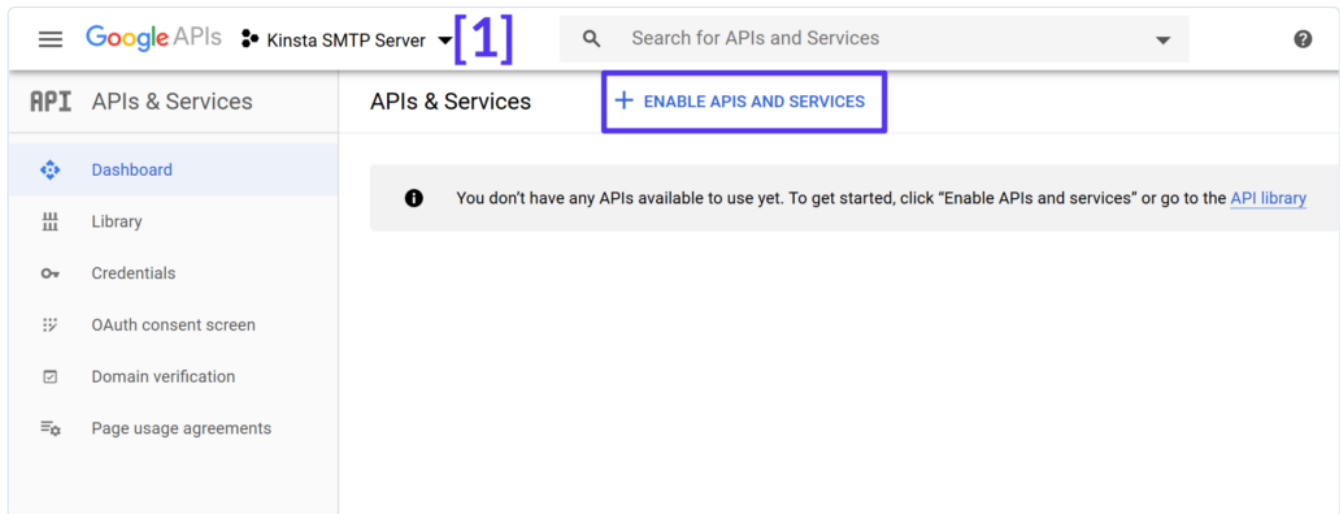
— Wähle die Gmail API Option

2. Erstelle deine Google App

Jetzt musst du eine Google-App erstellen. Damit kann deine WordPress-Seite sicher E-Mails über die Google Mail API versenden. Nochmals, dies ist definitiv der komplizierteste Teil dieses Prozesses. Hauptsächlich geht es aber darum, viele Buttons anzuklicken, und wir geben dir alle Schritte/Screenshots, die du benötigst, um durchzukommen.

Ein Projekt erstellen

Um anzufangen, öffne eine neue Registerkarte und [kopiere diese URL, um zur Google Developers Console zu gelangen](#). Dort musst du ein neues Projekt erstellen. Wenn du dich zum ersten Mal in die Developers Console einloggst, wird Google dich auffordern, dein erstes Projekt zu erstellen. Oder, wenn du bereits einige Projekte hast, kannst du ein neues Projekt erstellen, indem du auf das Dropdown-Menü oben links klickst (im Screenshot unten mit [1] gekennzeichnet).

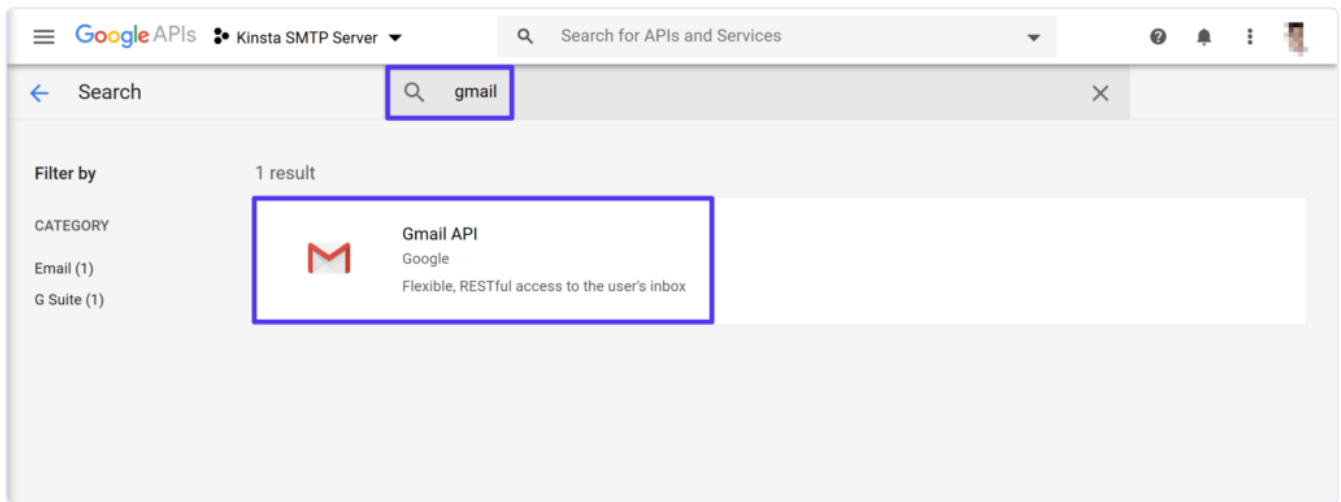


— Ein neues Google-Entwickler-Projekt erstellen

Aktiviere die Gmail API

Wenn du dein Projekt erstellt hast, klicke auf die Schaltfläche **Enable APIs and Services** (siehe Screenshot oben).

Auf dem nächsten Bildschirm suchst du nach „Gmail“ und wählst das **Gmail API** – Ergebnis aus:




— Suche nach der Gmail API

Dann klicke auf **Enable** auf der ganzen Gmail API Seite:

Google APIs Kinsta SMTP Server Search for APIs and Services

API Library



Gmail API

Google

Flexible, RESTful access to the user's inbox

ENABLE TRY THIS API ↗

Type
[APIs & services](#)

Last updated
12/10/19, 7:34 AM

Category
[Email](#)
[G Suite](#)

Service name
gmail.googleapis.com

Overview

The Gmail API lets you view and manage Gmail mailbox data like threads, messages, and labels.

About Google

Google's mission is to organize the world's information and make it universally accessible and useful. Through products and platforms like Search, Maps, Gmail, Android, Google Play, Chrome and YouTube, Google plays a meaningful role in the daily lives of billions of people.

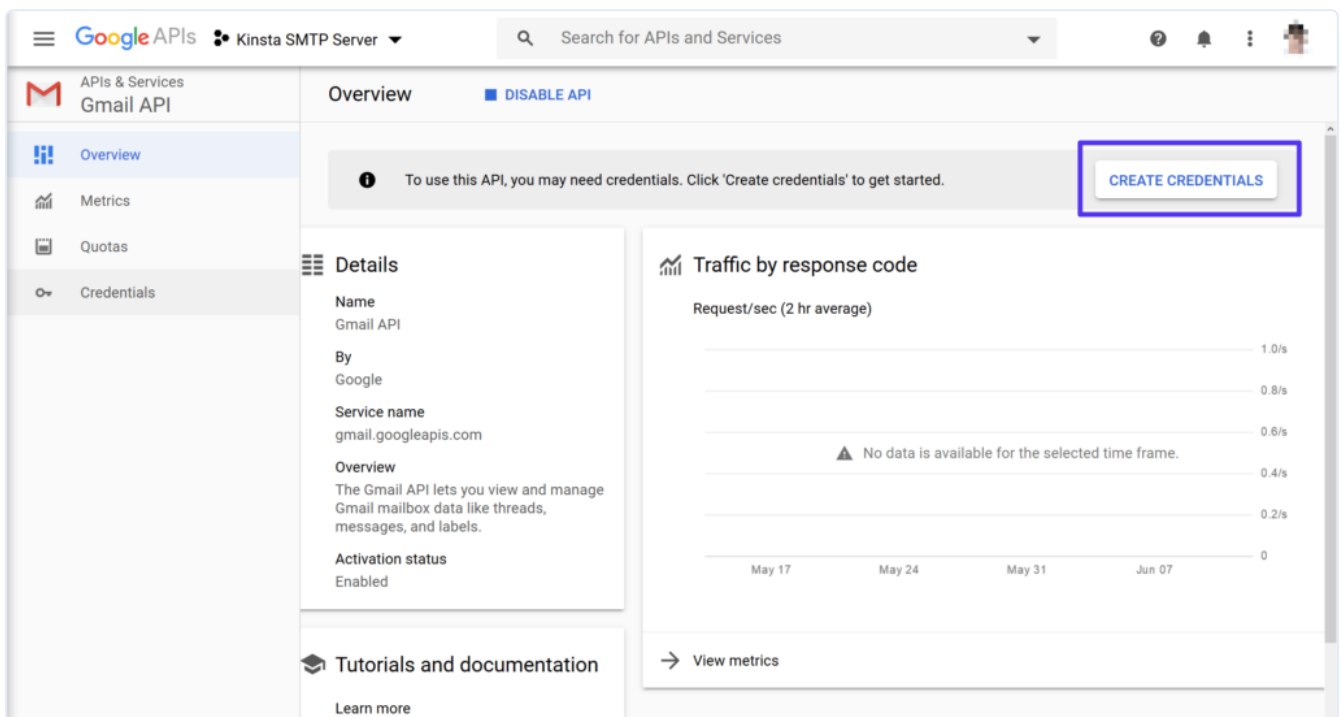
Tutorials and documentation

[Learn more](#) ↗

— Aktiviere die Gmail API

Anmeldeinformationen erstellen

Jetzt solltest du dich in einem speziell dafür vorgesehenen Interface für die Gmail API befinden. Klicke auf die Schaltfläche **Create Credentials**:



— Anmeldeinformationen für die Gmail API erstellen

Fülle das Formular **Find out what credentials you need** mit den folgenden Informationen aus:

- **Which API are you using?** Gmail API
- **Where will you be calling the API from?** Web browser (JavaScript)
- **What data will you be accessing?** User data

Wenn du das getan hast, klicke unten auf die Schaltfläche **What credentials do I need?**

Google APIs

Kinsta SMTP Server

Search for APIs and Services

API APIs & Services

Dashboard

Library

Credentials

OAuth consent screen

Domain verification

Page usage agreements

Credentials

Add credentials to your project

1 Find out what kind of credentials you need

We'll help you set up the correct credentials
If you wish you can skip this step and create an [API key](#), [client ID](#), or [service account](#)

Which API are you using?

Different APIs use different auth platforms and some credentials can be restricted to only call certain APIs.

Gmail API

Where will you be calling the API from?

Credentials can be restricted using details of the context from which they're called. Some credentials are unsafe to use in certain contexts.

Web browser (Javascript)

What data will you be accessing?

Different credentials are required to authorize access depending on the type of data that you request.

☒ User data

Access data belonging to a Google user, with their permission

☐ Application data

Access data belonging to your own application

What credentials do I need?

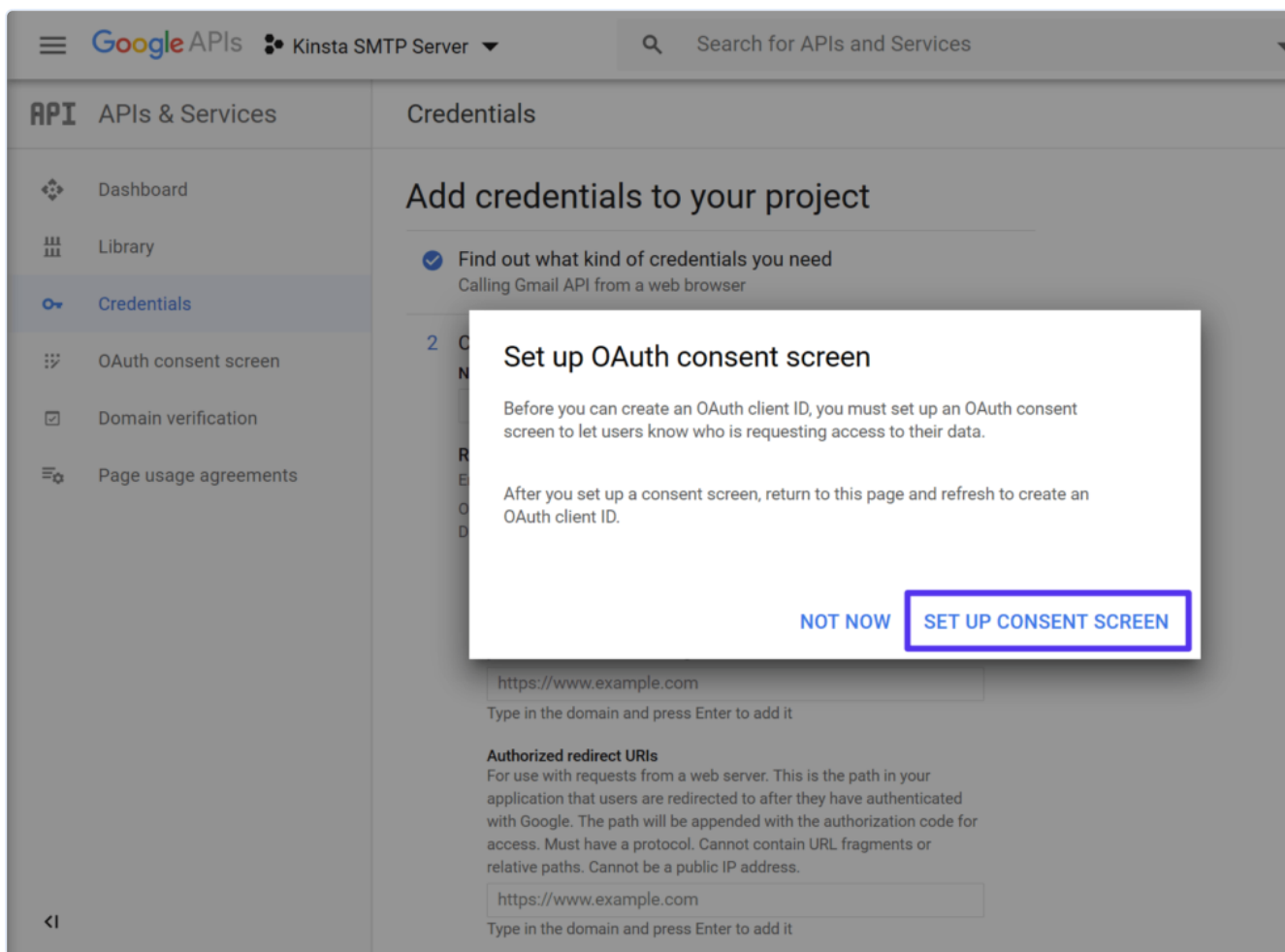
— Anmeldeformular ausfüllen

Einen Zustimmungsbildschirm einrichten

Nun wird Google dich auffordern, einen Zustimmungsbildschirm zu erstellen. Dies ist die gleiche Art von Autorisierungsbildschirm, die du siehst, wenn du Google benutzt, um dich auf einer Seite anzumelden/einzuloggen.

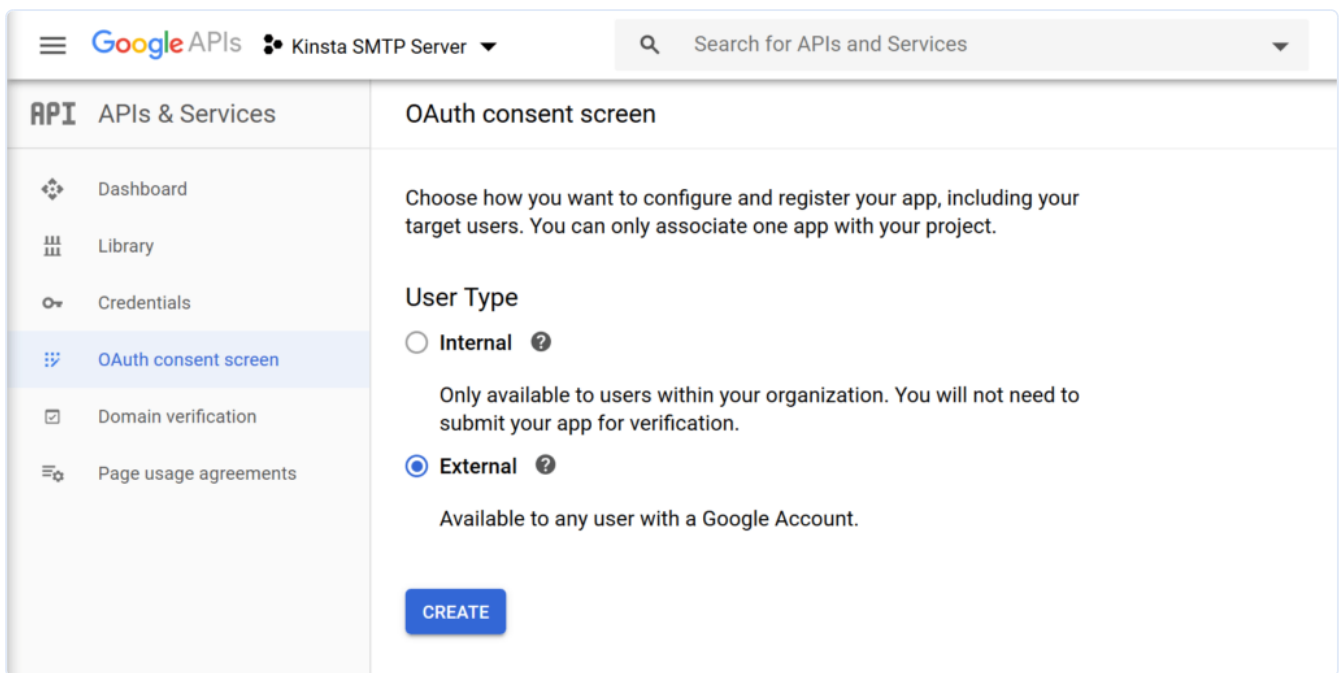
Du musst dies tun, um Googles Anforderungen zu erfüllen, aber du brauchst dir keine Gedanken über die Informationen zu machen, die du eingibst, da du diese nur für deine eigene WordPress-Seite verwenden wirst.

Klicke auf die Schaltfläche **Set Up Consent Screen**:



— Die Eingabeaufforderung zum Erstellen eines OAuth Zustimmungsbildschirm

Dies sollte einen neuen Reiter öffnen, in dem du einen **OAuth consent screen** konfigurieren kannst (behalte den ursprünglichen Browser-Reiter griffbereit, da du in einer Sekunde wieder dorthin zurückgehen musst). Unter **User Type** wähle **External**. Klicke dann auf **Create**:



— Erstelle einen externen Zustimmungsbildschirm

Auf dem nächsten Bildschirmfenster gibst du grundlegende Details für deine Seite ein. Nochmals – du brauchst dir keine Gedanken darüber zu machen, was du hier einträgst, denn du bist die einzige Person, die diese Informationen sehen wird.

The screenshot shows the 'OAuth consent screen' configuration page in the Google APIs console. The left sidebar contains a menu with 'Dashboard', 'Library', 'Credentials', 'OAuth consent screen' (highlighted), 'Domain verification', and 'Page usage agreements'. The main content area is titled 'OAuth consent screen' and includes a description: 'Before your users authenticate, this consent screen will allow them to choose whether they want to grant access to their private data, as well as give them a link to your terms of service and privacy policy. This page configures the consent screen for all applications in this project.' Below this, the 'Verification status' is 'Not published'. The 'Application name' is 'Kinsta Gmail SMTP Server'. The 'Application logo' section has a 'Local file for upload' button and a 'Browse' button. The 'Support email' is 'cm001@gmail.com'. The 'Scopes for Google APIs' section explains that scopes allow access to user data and provides a link to 'Learn more'. On the right, there are sections for 'About the consent screen' and 'OAuth verification', which lists conditions for verification: sensitive/restricted scope, app icon, many domains, or changes to a previously-verified screen. A link at the bottom right says 'Let us know what you think about our OAuth'.










— Konfiguriere den Zustimmungsbildschirm

Sobald du die Informationen hinzugefügt hast, klicke unten auf **Save**.

Beende das Hinzufügen der Anmeldeinformationen zu deinem Projekt

Geh nun zurück zu dem Reiter, der **Add credentials to your project interface** beinhaltet und gib die folgenden Informationen ein:

- **Name** – der Name deiner Webseite (oder etwas anderes, das man sich leicht merken kann).
- **Authorized JavaScript origins** – du kannst dies im Post SMTP Mailer/Email Log Plugin finden (Schritt #1).
- **Authorized redirect URIs** – du findest dies im Post SMTP Mailer/Email Log Plugin (Schritt #1).

API APIs & Services	Credentials
<div><div> Dashboard</div><div> Library</div><div> Credentials</div><div> OAuth consent screen</div><div> Domain verification</div><div> Page usage agreements</div></div>	<h2>Add credentials to your project</h2> <div><div> Find out what kind of credentials you need</div><div>Calling Gmail API from a web browser</div></div> <div><div>2</div>Create an OAuth 2.0 client ID</div> <div><div>Name </div><div>Kinsta WordPress Site</div></div> <div><div>Restrictions</div><div>Enter JavaScript origins, redirect URIs, or both Learn More</div><div>Origins and redirect domains must be added to the list of Authorized Domains in the OAuth consent settings.</div><div><div>Authorized JavaScript origins</div><div>For use with requests from a browser. This is the origin URI of the client application. It can't contain a wildcard (https://*.example.com) or a path (https://example.com/subdir). If you're using a nonstandard port, you must include it in the origin URI.</div><div>https://test.kinsta.cloud</div><div>Type in the domain and press Enter to add it</div><div><div>Authorized redirect URIs</div><div>For use with requests from a web server. This is the path in your application that users are redirected to after they have authenticated with Google. The path will be appended with the authorization code for access. Must have a protocol. Cannot contain URL fragments or relative paths. Cannot be a public IP address.</div><div>st.kinsta.cloud/wp-admin/options-general.php?page=postman</div><div>Type in the domain and press Enter to add it</div></div><div><div>Refresh </div></div></div></div>

— Erstelle deinen Anmeldeinformationen

Sobald du alles hinzugefügt hast, klicke auf **Refresh**. Dann sollte sich die Schaltfläche **Refresh** zu **Create OAuth client ID** ändern – klicke darauf, um den Vorgang abzuschließen und klicke dann auf **Done**.

Jetzt bist du fast fertig!

Sobald du auf **Done** klickst, solltest du einen Abschnitt für **OAuth 2.0-Client-IDs** auf der Registerkarte **Credentials** deines Projekts sehen (der sich automatisch öffnen sollte, nachdem du auf **Done** geklickt hast).

Klicke auf den Eintrag für den Namen, den du gerade erstellt hast:

The screenshot shows the Google Cloud 'Credentials' page. The left sidebar has 'APIs & Services' with 'Credentials' selected. The main area has a 'Credentials' header with '+ CREATE CREDENTIALS' and 'DELETE' buttons. Below is a message: 'Create credentials to access your enabled APIs. [Learn more](#)'. The 'API Keys' section is empty, showing 'No API keys to display'. The 'OAuth 2.0 Client IDs' section contains one entry, 'Kinsta WordPress Site', which is highlighted with a red box. The entry details are: Name: Kinsta WordPress Site, Creation date: Jun 12, 2020, Type: Web application, Client ID: 736389733285-f55t... The 'Service Accounts' section is empty, showing 'No service accounts to display'.

<input type="checkbox"/>	Name	Creation date ↓	Restrictions	Key	Usage with all services (last 30 days) ?
No API keys to display					

<input type="checkbox"/>	Name	Creation date ↓	Type	Client ID	
<input type="checkbox"/>	Kinsta WordPress Site	Jun 12, 2020	Web application	736389733285-f55t...	

<input type="checkbox"/>	Email	Name ↑	Usage with all services (last 30 days) ?
No service accounts to display			

— Zugang zu OAuth 2.0 Client-IDs

Jetzt brauchst du nur noch zwei verschiedene Informationen zu finden:

- Client ID
- Client secret

Bewahre diese Werte zugänglich auf, denn du wirst sie im nächsten Schritt brauchen:

[←](#) Client ID for Web application [↓ DOWNLOAD JSON](#) [↻ RESET SECRET](#) [🗑 DELETE](#)

Name *
Kinsta WordPress Site

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

Client ID
73C...
f55...ivoe.ap
ps.googleusercontent.com

Client secret
3_Z...Eb

Creation date
June 12, 2020 at 9:20:24 AM GMT+7

i

The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorized domains](#).

Authorized JavaScript origins **?**
For use with requests from a browser

[+ ADD URI](#)

— Deine Gmail API Client IDs

3. Fügen Gmail App Client IDs zum Post SMTP Mailer/Email Log hinzu

Um das Setup abzuschließen, geh zurück zu den Post SMTP Mailer/Email Log Einstellungen in deinem WordPress Dashboard und füge die **Client ID** und das **Client Secret** aus dem vorherigen Schritt ein. Stelle dann sicher, dass du deine Änderungen speicherst:

Transport

Choose SMTP or a vendor-specific API:

Type

Gmail API

Mailer Type

PostSMTP

*Beta Feature: Change this to **PHPMailer** only if you see `wp_mail` conflict message, conflicts when another plugin is activated, and *sometimes* spam issues.*

Authentication

Attention! Check this article how to configure Gmail/Gsuite OAuth: [Read Here](#)

Authorized JavaScript origins

https://test.kinsta.cloud

Authorized redirect URI

https://test.kinsta.cloud/wp-admin/options-general.php?page=postman

Client ID

2416...0.apps.googleusercontent.c

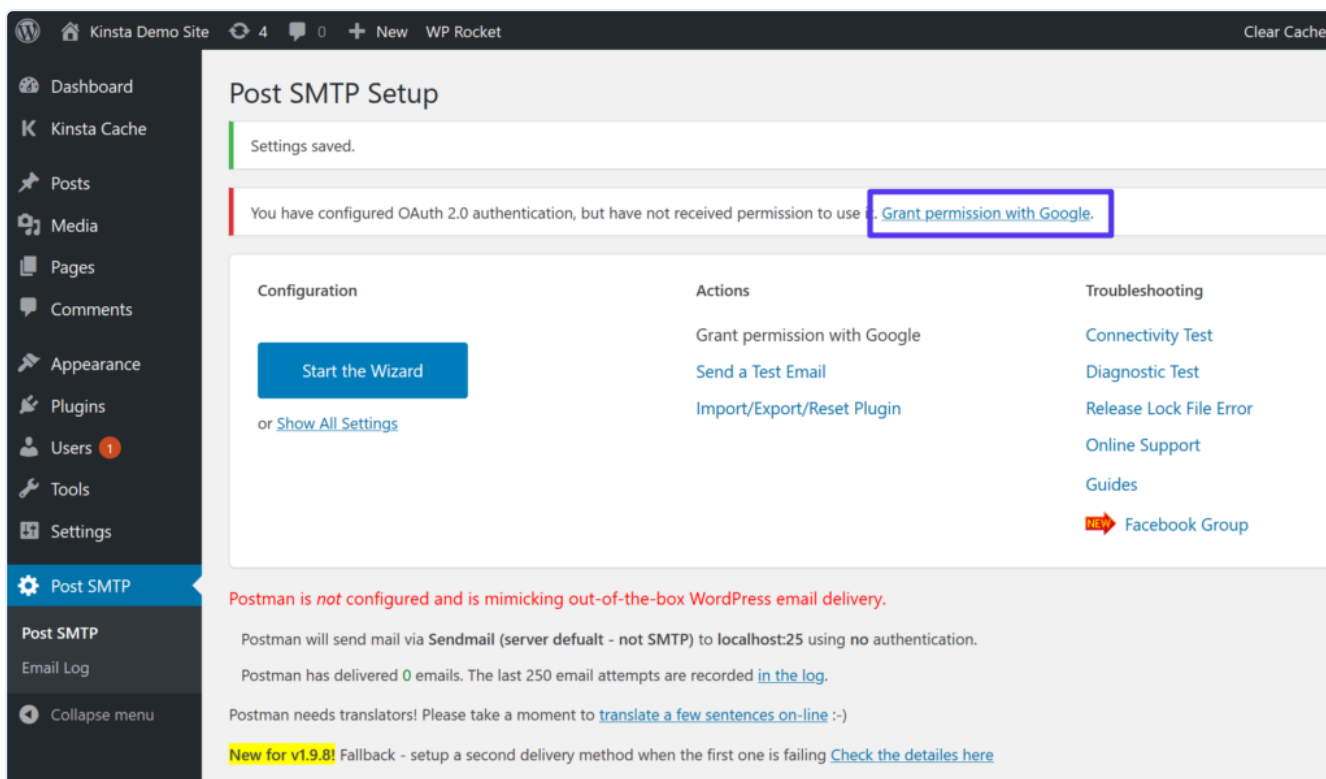
Client Secret

B8...Af

Save Changes

— Gmail API Client IDs zu WordPress hinzufügen

Dann sollte das Plugin dich folgendes auffordern: **Grant permission with Google:**



— Erlaubnis für Google erteilen

Wenn du auf diesen Link klickst, öffnet sich der normale Google-Autorisierungsprozess (wieder so, als ob du dich für eine Seite mit Google-Anmeldung registrieren würdest). Da du deine App jedoch nicht zur Überprüfung bei Google eingereicht hast, zeigt Google dir eine Warnung an: „This app isn’t verified“.

Da dies deine eigene App ist, kannst du die Warnung getrost ignorieren. Klicke auf die Option, erweiterte Einstellungen anzuzeigen und klicke dann auf den Link **Go to “yourwebsite.com” (unsafe)**, um den Autorisierungsprozess fortzusetzen:



This app isn't verified

This app hasn't been verified by Google yet. Only proceed if you know and trust the developer.

If you're the developer, submit a verification request to remove this screen. [Learn more](#)

[Hide Advanced](#)

[BACK TO SAFETY](#)

Google hasn't reviewed this app yet and can't confirm it's authentic. Unverified apps may pose a threat to your personal data. [Learn more](#)

[Go to kinsta.cloud \(unsafe\)](#)

— Ignoriere die Warnung fortzufahren

Du wirst nun den regulären Prozess durchlaufen. Vergewissere dich, dass du die Option **Allow** wählst, um deiner WordPress-Seite Zugang zu deinem Gmail-Konto zu gewähren. Sie benötigt diese Berechtigungen, um E-Mails über den Gmail SMTP-Server zu versenden.

Und das ist alles! Es gab ziemlich viele Schritte, aber jetzt bist du so gut wie fertig.

4. Sende eine Test-E-Mail

Um sicherzugehen, dass alles funktioniert, enthält der Post SMTP Mailer/Email Log eine Option zum Versenden einer Test-Email. Du kannst diese von der Haupteinstellungsseite aus aufrufen:

Post SMTP Setup

Configuration	Actions	Troubleshooting
Start the Wizard or Show All Settings	Grant permission with Google Send a Test Email Import/Export/Reset Plugin	Connectivity Test Diagnostic Test Release Lock File Error Online Support Guides Facebook Group

Postman is configured.

Postman will send mail via the **Gmail API**.

Postman has delivered **0** emails. The last 250 email attempts are recorded [in the log](#).

Postman needs translators! Please take a moment to [translate a few sentences on-line :-\)](#)

New for v1.9.8! Fallback - setup a second delivery method when the first one is failing [Check the details here](#)

— Wie man eine Test-E-Mail über die Gmail API versendet

Du kannst die E-Mail eingeben, an die du eine Testnachricht senden möchtest.

Dann solltest du in den Einstellungen des Plugins eine Erfolgsmeldung sehen:

Post SMTP Setup

Send a Test Email

[+ Back To Main Menu](#)

1. Specify the Recipient

2. Send The Message

3. Session Transcript

Sending the message: **Success**

Status

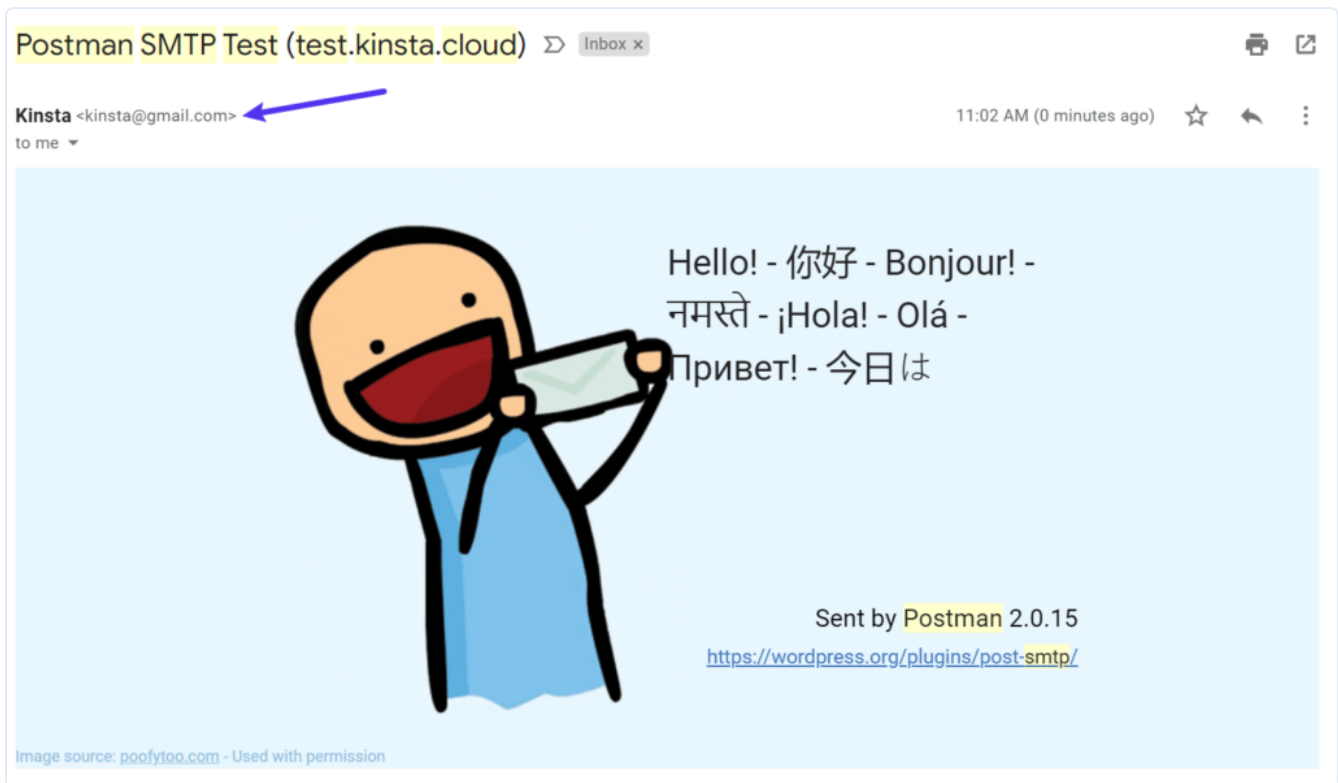
Your message was delivered (1140 ms) to the SMTP server!
Congratulations :)

Next

Finish

— Die Erfolgsnachricht für deine Test-E-Mail

Und wenn du zu deinem E-Mail-Posteingang gehst, solltest du auch eine Test-E-Mail sehen:



— Die eigentliche E-Mail, die du in deinem Posteingang sehen solltest

Wenn alles gut geht, bist du fertig.

Deine WordPress-Seite wird nun alle E-Mails über den Gmail SMTP-Server versenden. Du kannst sicherstellen, dass dies geschieht, indem du zum **Post SMTP ? Email Log** gehst. Hier werden alle E-Mails aufgelistet, die deine Seite versendet (zusammen mit eventuellen Fehlern, falls das Plugin auf Probleme stößt):

Dashboard

Kinsta Cache

Posts

Media

Pages

Comments

Appearance

Plugins 1

Users

Tools

Settings

Post SMTP

Post SMTP

Email Log

Collapse menu

Post SMTP Email Log

Please notice: when you select a date for example 11/20/2017, behind the scene the query select 11/20/2017 00:00:00.
So if you searching for an email arrived that day at any hour you need to select 11/20/2017 as the **From Date** and 11/21/2017 as the **To Date**.

This is a record of deliveries made to the mail server. It does not necessarily indicate successful delivery to the recipient.

From Date

From Date

To Date

To Date

Search

Search

Records per page

10

Filter/Search

Export To CSV

Trash All

Bulk actions

Apply

2 items

<input type="checkbox"/>	Subject	Sent To	Status	Solution	Delivery Time
<input type="checkbox"/>	Postman SMTP Test (test.kinsta.cloud)	test@kinsta.com	Sent	All good, mail sent.	October 22, 2020 4:14 am
<input type="checkbox"/>	Postman SMTP Test (test.kinsta.cloud)	test@kinsta.com	Sent	All good, mail sent.	October 22, 2020 4:02 am

Bulk actions

Apply

2 items

— Wie man ein Protokoll der E-Mails sehen kann, welches deine Seite versendet

Zusammenfassung

Mit dem Gmail SMTP-Server kannst du E-Mails über dein Gmail-Konto und Googles Server versenden.

Eine Möglichkeit besteht darin, E-Mail-Clients von Drittanbietern, wie Thunderbird oder Outlook, so zu konfigurieren, dass sie E-Mails über dein Gmail-Konto versenden. Die Standard Gmail SMTP Details sind wie folgt:

- **Gmail SMTP-Serveradresse:** smtp.gmail.com
- **Gmail SMTP Name:** Dein voller Name
- **Gmail SMTP-Benutzername:** Deine vollständige Gmail-Adresse (z.B. you@gmail.com)
- **Gmail SMTP Passwort:** Das Passwort, mit dem du dich bei Gmail einloggst
- **Gmail SMTP-Port (TLS):** 587
- **Gmail SMTP-Port (SSL):** 465

Eine andere Möglichkeit ist, Gmail zu benutzen, um die Transaktions-E-Mails deiner WordPress-Seite zu versenden. Mit einem Limit von 500 E-Mails pro Tag ist das Limit für den kostenlosen Versand von Gmail deutlich höher als bei [anderen kostenlosen SMTP-Diensten](#) wie [SendGrid](#) oder [Mailgun](#).

Wenn du dies jedoch tust, solltest du E-Mails über die Gmail-API versenden, anstatt nur die SMTP-Serverdetails zu verwenden.

Auch wenn das Einrichten einer App für die Nutzung der Gmail API den Prozess einmal komplizierter macht, ist es die Mühe wert, denn es gibt dir einen zuverlässigen und sicheren Weg, die E-Mails deiner Seite zu versenden.

Nun, da du alles vorbereitet hast, schau dir unsere Liste von [Gmail Add-Ons](#) an, um deine E-Mail-Produktivität zu verbessern.

Hast du noch Fragen über den Gmail SMTP Server oder wie du ihn mit WordPress nutzen kannst? Frag uns in den Kommentaren!