

MailXstream Zapier Integration Guide

Version: 1.0 | **Last Updated:** August 2026 | **Zapier App Version:** 2.0.0

What is the MailXstream Zapier Integration?

Zapier connects MailXstream to more than 7,000 other applications without writing any code. A **Zap** is an automated workflow: something happens in one app (the **trigger**), and Zapier makes something happen in another (the **action**).

With the MailXstream integration you can:

- **Mail automatically** — a document lands in Google Drive, Dropbox, or your accounting system, and MailXstream prints and mails it.
- **Pull addresses from anywhere** — recipient details come from a spreadsheet, CRM, or database at the moment the order is created.
- **Stay informed** — post to Slack, update a record, or send an email when an order completes, errors, or needs approval.
- **Look orders up** — check an order's state from inside any Zap.

Everything the integration does is something you could do by hand in MailXstream. Zapier removes the manual step.

About This Guide

Purpose

This guide explains how to connect MailXstream to Zapier, how to build each supported kind of Zap, and — in the most detail — how to supply the **delivery address** and **mass mailing address file** that many templates require. Those two need extra setup, because they are per-order data that must be gathered when the Zap runs.

Intended Audience

This guide is for MailXstream users who want to automate order submission.

Your MailXstream role still applies when you use Zapier. A Zap acts as you — it can do exactly what your own account can do, no more.

Role	Can build order-submitting Zaps?
User	Yes — submit orders and monitor status
Monitor	No — read-only; can use triggers and Find Order, but not Send Document
Manager	Yes — all User capabilities
Admin	Yes — all Manager capabilities

Prerequisites

Before you begin, make sure you have:

- A MailXstream user account with permission to create orders.
- A Zapier account. Free plans work, but multi-step Zaps — which most of the scenarios in this guide use — require a paid Zapier plan.
- The **MailXstream** app added to that Zapier account. It is a private app, so it does not appear in a Zapier search until you add it — see [Adding the MailXstream App to Zapier](#).
- An **access token** created on your MailXstream profile page (see [Creating an Access Token](#)).
- Knowledge of which **template** you want to submit against, and whether it needs a delivery address or an address file (see [Knowing What Your Template Needs](#)).

Conventions Used in This Guide

Convention	Meaning
Bold	A button, field label, or menu item you interact with
<code>Code</code>	A value you type exactly as shown
> Tip	Advice that saves time
> Important	Something that will cause problems if missed

Quick Start

Send Your First Automated Document in 7 Steps

1. **Add the MailXstream app to Zapier** using the invite link on the MailXstream **Support** page — it is a private app, so this comes first
2. **Create an access token** in MailXstream (Profile → User Tokens → Create User Token)
3. **Create a Zap** in Zapier and choose your trigger — for example, *Google Drive → New File in Folder*
4. **Add an action** and search for **MailXstream**, then choose **Send Document**
5. **Connect your account** — choose your environment (Production or Test) and paste your token
6. **Choose a template** — the form then expands to show that template's own options
7. **Map the document** from your trigger into the **Document** field, fill in anything required, and **Test** the step

For detailed instructions, see [Connecting MailXstream to Zapier](#).

What the Integration Provides

Type	Name	What it does
Action	Send Document	Submits a document to be printed and mailed
Trigger	Order Completed	Fires when an order finishes printing and mailing
Trigger	Order Errored	Fires when an order hits an error
Trigger	Order Action Required	Fires when an order is waiting for approval
Search	Find Order	Looks up an order by name and returns its state

Connecting MailXstream to Zapier

Adding the MailXstream App to Zapier

MailXstream is a **private** Zapier app. It is not listed in Zapier's public app directory, and searching for it inside the Zap editor finds nothing until it has been added to your Zapier account.

Step 1. In MailXstream, open **Support** from the left-hand menu.

Step 2. Under **User Guides**, click **Add MailXstream to Zapier**.

Step 3. Sign in to Zapier if you are prompted, then accept the invitation.

You only need to do this once per Zapier account. Afterwards, **MailXstream** appears in the app search whenever you add a step to a Zap.

Tip: Accept the invitation while signed in to the same Zapier account you build your Zaps with. If you have both a personal and a work Zapier account, it is easy to add the app to the wrong one.

Understanding Environments

MailXstream runs two separate environments, and **each issues its own access tokens**.

Environment	Purpose	Orders are actually mailed?
Production	Your live account	Yes — real printing, real postage
Test	Sandbox for building and proving a Zap	No

Build and prove new Zaps against **Test** first, then switch to Production once you are confident. You can hold a connection to each at the same time — Zapier lists them separately, and each connection's name ends with its environment, for example

Jane Doe @ ACME Corp (Test) .

Important: A Production token will not work on a Test connection, or vice versa. Create the token on the same environment you select in Zapier.

Step 1: Create an Access Token

Step 1. Log in to MailXstream on the environment you want to connect.

Step 2. Click your name in the top-right corner and choose **Profile**.

Step 3. Scroll to the **User Tokens** panel and click **Create User Token**.

Step 4. Enter a **Token Name** that says where it is used — for example `Zapier – Invoice Automation`.

Step 5. Optionally set an expiry in days. Leave blank for a token that does not expire.

Step 6. Click **Generate**, then use the copy button to copy the token.

Important: The token is shown **once**. Copy it before closing the dialog. If you lose it, delete the token and create a new one.

Tip: Use a separate token for each Zap or system. If one is ever compromised, you can revoke just that one without breaking everything else.

Step 2: Connect the Account in Zapier

Step 1. In your Zap, add a **MailXstream** step and choose an action or trigger.

Step 2. Click **Sign in** on the Account panel.

Step 3. In the pop-up window, choose your **Environment** — Production or Test.

Step 4. Paste your **Access Token**.

Step 5. Click **Yes, Continue**.

Zapier verifies the token immediately. On success, the connection appears with your name, client, and environment.

How Send Document Works

Understanding this section makes everything else straightforward.

Setup Time vs. Run Time

A Zap has two moments that matter:

Moment	What happens	Example
Setup time	You build the Zap	Choosing the template
Run time	The Zap fires, once per trigger event	The actual document and recipient address

The only field fixed at setup time is the **Template**, because choosing it changes the rest of the form. Everything else can be either typed in (the same on every run) or **mapped** from an earlier step (different on every run).

Mapping Data From Earlier Steps

Mapping is how run-time data reaches MailXstream. Click any field and Zapier shows the data available from earlier steps; pick a value and Zapier inserts it as a token. When the Zap runs, the token is replaced by that run's real value.

You can combine mapped values with typed text in the same field — this matters for addresses, covered next.

The Template Drives the Form

When you pick a template, Zapier asks MailXstream what that template needs and rebuilds the form. Two templates can therefore show completely different fields.

Always present:

Field	Required	Notes
Template	Yes	Choosing it reshapes the fields below
Document	Yes	The PDF to print and mail
Order Name	No	Defaults to auto — MailXstream generates a unique name

Added by the template, if it defines them: Delivery Address, Return Address, Address File, Ink, Plex, and any other option the template offers.

Important: If you change the template later, the fields below it change too, and any mappings you had set on the old fields are lost. Re-check every field after switching templates.

Knowing What Your Template Needs

The Zap editor tells you directly: after choosing a template, any field marked **required** (a red asterisk) must be filled before the Zap will run.

You can also check in MailXstream itself — start creating an order by hand with that template and look at which options appear. What you see there is what Zapier will show you.

If the template shows...	Then in Zapier you will see...	Covered in
A delivery address box	Delivery Address — a multi-line text field	Scenario 2
An address file upload	Address File — a file field	Scenario 4
Neither	Just Template, Document, Order Name	Scenario 1

Order Names

Leave **Order Name** as and MailXstream generates a unique name. Supply your own and it must be unique — a duplicate fails with *"Order name is already used."*

Tip: If you want meaningful names, map something that varies per run — an invoice number, a filename, a timestamp. A typed constant like `Daily Invoices` will work on the first run and fail on every one after.

Supplying a Delivery Address

Templates for single documents often need to be told where to mail the piece. The **Delivery Address** field is a multi-line box: **one address line per row**, exactly as it should appear on the envelope.

```
Jane Smith  
ACME Corporation  
123 Main Street, Suite 400  
Springfield, IL 62704
```

There are three ways to fill it.

Method 1: Type a Fixed Address

Use when every run goes to the same place — a compliance office, a records department.

Type the address into the field, pressing Enter between lines.

Method 2: Map From the Trigger

Use when the trigger already carries the address — a form submission, a CRM record, a new spreadsheet row.

Click into **Delivery Address** and insert one field per line:

```
1. First Name 1. Last Name  
1. Company  
1. Street Address  
1. City, 1. State 1. Zip
```

Tip: Put the literal comma and spaces between `City`, `State`, and `Zip` yourself — mapped values are inserted exactly as stored, with no punctuation added.

Method 3: Look the Address Up Mid-Zap

This is the most common case, and the one that needs an extra step. The trigger gives you a document but no address — the address lives somewhere else, keyed by something in the document or its filename.

Insert a **lookup step between the trigger and MailXstream:**

Step 1. Add a step after your trigger — for example **Google Sheets → Lookup Spreadsheet Row**, **Airtable → Find Record**, or **Webhooks by Zapier → GET**.

Step 2. Tell it what to match on. If your files are named like `INV-10432.pdf`, match the sheet's invoice column against the trigger's filename.

Step 3. In the MailXstream step, map the **looked-up** fields — from step 2, not step 1 — into **Delivery Address**, one per line.

A complete example:

Step	App	Configuration
1	Google Drive	New File in Folder — watches <code>/Invoices/Outgoing</code>
2	Formatter	Text → Extract Pattern — pulls <code>INV-10432</code> from the filename
3	Google Sheets	Lookup Spreadsheet Row — finds the row where Invoice = step 2's output
4	MailXstream	Send Document — Document from step 1, Delivery Address from step 3

Tip: Step 2 is only needed when the key is buried inside a longer filename. If the file is simply named `INV-10432.pdf`, Google Sheets can often match on the trigger's filename directly.

Verifying the Address Is Right

Run the Zap's test and look at the **Data Out** panel — it shows exactly what was sent. Check that lines are separate rather than run together, and that no line is blank because a lookup returned nothing.

Important: A lookup that finds no match usually returns empty values rather than failing. The order is then submitted with a partial address. If your lookup can miss, add a **Filter by Zapier** step after it that only continues when the address field is not empty.

Supplying a Mass Mailing Address File

Mass mailing templates take **one document** and mail it to **many recipients**, using an uploaded address list. In Zapier this appears as a **file field**, usually labelled **Address File**.

What the File Must Contain

The format is defined by your template, not by Zapier — it is the same file you would upload by hand in MailXstream. Confirm the expected columns and order with your MailXstream administrator, and test with a short list before running a large one.

Method 1: Map a File From Another App

Use when the list already exists as a file.

Step 1. Add a step that produces the file — **Google Drive → Find a File**, **Dropbox → Find a File**, or an email trigger with an attachment.

Step 2. In the MailXstream step, click **Address File** and map the file object from that step.

Any Zapier field that produces a file works here. Both the document and the address file can come from the same trigger — for example an email with two attachments.

Method 2: Build the File From Data

Use when the addresses live in spreadsheet rows or database records rather than an existing file.

A file field needs a file. Text alone will not do — you must create a file first, then map it.

Step 1. Gather the rows. **Looping by Zapier**, or an app step that returns line items, collects the recipients.

Step 2. Format them as CSV text — **Formatter** → **Utilities** → **Line Items to Text** with a newline separator.

Step 3. Write that text to a file: **Google Drive** → **Upload File** (or Dropbox) with the CSV text as the content and a name ending in `.csv`.

Step 4. Map the file produced by step 3 into **Address File**.

Important: This is genuinely more work than the single-document path, and it is worth asking whether you need it. If your recipient list is stable, keeping one maintained CSV in Drive and mapping it with Method 1 is far simpler than rebuilding it on every run.

Choosing Between Mass Mail and One Order Per Recipient

There are two ways to mail many people, and they are not equivalent:

Approach	How	Result	Best when
Mass mailing	One Send Document with an address file	One order , many pieces	The same document goes to everyone
One order each	Trigger fires per recipient, Delivery Address mapped	One order per recipient	Each recipient gets a different document

Mass mailing is more efficient and gives you a single order to track. Per-recipient orders are the only option when the documents differ — personalized statements, individual invoices.

Tip: Zapier plans have task limits, and one order per recipient consumes a task per recipient. For large volumes, mass mailing is dramatically cheaper.

Scenarios

Each scenario is a complete, working Zap. Start from whichever is closest to your situation.

Scenario 1: Simple Document, No Extra Data

Use when: the template already knows where to mail — the address is printed on the document itself, or the template has a fixed recipient.

Step	App	Configuration
1	Google Drive	New File in Folder — watches your outgoing folder
2	MailXstream	Send Document — Template: your template; Document: 1. File ; Order Name: auto

This is the simplest possible Zap and needs no lookup at all.

Scenario 2: Single Document With a Looked-Up Address

Use when: the document arrives without an address, and the address lives in a spreadsheet or CRM.

Step	App	Configuration
1	Google Drive	New File in Folder
2	Google Sheets	Lookup Spreadsheet Row — match on the filename
3	MailXstream	Send Document — Document: 1. File ; Delivery Address: mapped from step 2, one field per line

See [Method 3](#) for detail. Add a **Filter** step between 2 and 3 if the lookup can miss.

Scenario 3: Document Generated by Another App

Use when: the PDF does not exist yet — it is produced by your accounting, HR, or document system.

Step	App	Configuration
1	QuickBooks / Xero / Stripe	New Invoice (or equivalent)
2	Google Docs / PDF app	Create Document from Template — produces the PDF
3	MailXstream	Send Document — Document: from step 2; Delivery Address: from step 1's customer fields

The address usually comes free here — the record that triggered the Zap already has it.

Scenario 4: Mass Mailing With an Address File

Use when: one document goes to many recipients.

Step	App	Configuration
1	Schedule by Zapier	Every Month (or any trigger you prefer)
2	Google Drive	Find a File — the current newsletter PDF
3	Google Drive	Find a File — the recipient list CSV
4	MailXstream	Send Document — Document: step 2; Address File: step 3

See [Supplying a Mass Mailing Address File](#) if the list must be built rather than found.

Scenario 5: Notify a Team When an Order Completes

Use when: people need to know that mail went out — without logging in to check.

Step	App	Configuration
1	MailXstream	Order Completed
2	Slack	Send Channel Message — include <code>1. Order Name</code> and <code>1. Order ID</code>

Swap Slack for Teams, email, or a spreadsheet row. The same shape works for any notification.

Scenario 6: Escalate Errors and Approvals

Use when: stuck orders must not sit unnoticed.

Step	App	Configuration
1	MailXstream	Order Errored — or Order Action Required
2	Slack / Email	Alert the responsible team
3	Jira / Asana (optional)	Create Issue so the follow-up is tracked

Tip: Build **Order Action Required** early. An order waiting for approval is invisible until someone looks, and this is the Zap that makes it visible.

Scenario 7: Log Every Order to a Spreadsheet

Use when: you want a running record outside MailXstream for reporting or audit.

Step	App	Configuration
1	MailXstream	Order Completed
2	Google Sheets	Create Spreadsheet Row — one column per output field

Scenario 8: Check an Order From Another Workflow

Use when: a different system needs to know an order's state.

Step	App	Configuration
1	Any trigger	For example a Slack command or form submission
2	MailXstream	Find Order — Order Name: mapped from step 1
3	Any action	Reply with <code>2. State</code> and <code>2. Status</code>

Find Order supports wildcards: `Invoice*` matches every order whose name starts with `Invoice`. It returns up to 10 matches, most recently changed first.

Scenario 9: Submit Only When Conditions Are Met

Use when: not everything that triggers should be mailed.

Step	App	Configuration
1	Google Drive	New File in Folder
2	Filter by Zapier	Only continue if — filename ends with <code>.pdf</code> , amount exceeds a threshold, etc.
3	MailXstream	Send Document

Tip: A filter is the cheapest safeguard in Zapier. When a Zap will submit real mail, a filter that stops the wrong runs is worth more than any amount of care downstream.

Understanding Triggers

How Polling Works

MailXstream triggers **poll** — Zapier checks for new orders on a schedule set by your Zapier plan, typically every 1-15 minutes. Triggers are not instant, and an order may take a few minutes to appear.

Each trigger fires **once** per order reaching that condition, and will not re-fire for the same event on later checks.

Trigger Reference

Trigger	Fires when	Typical use
Order Completed	Fully processed, printed, and mailed	Notify, log, close the loop
Order Errored	An error needs attention	Alert someone who can fix it
Order Action Required	Waiting for manual approval	Prompt the approver

Data Each Trigger Provides

Field	Description
Order ID	Unique identifier, e.g. <code>ORD_17563068339779DEV</code>
Order Name	The order's name
Template Name	Template it was submitted against
State	Current stage — Segment, Print, Mail, Complete
Status	Within that stage — Wait, Run, Complete, Error
Last State Change	Unix timestamp of the last change
Has Errors	True if the order has errors
Action Required	True if waiting for approval
Test Order	True if submitted as a test

Tip: **Last State Change** is a Unix timestamp. Use **Formatter** → **Date/Time** to convert it to something readable before putting it in a message.

Testing Your Zap

Step 1. Build against Test. Connect a Test-environment account first. Orders submitted there are not printed or mailed.

Step 2. Use the step test. Zapier's **Test** button runs the step for real and shows what was sent and received.

Step 3. Read Data Out. Confirm the address lines are separate, the right file was attached, and no mapped field is empty.

Step 4. Confirm in MailXstream. Log in and find the order. It should look exactly like one you created by hand.

Step 5. Switch to Production. Change the connection on the MailXstream step, re-test once, then turn the Zap on.

Important: Once the Zap is on and connected to Production, every trigger event mails for real. Watch the first few live runs in **Zap History** rather than assuming success.

Troubleshooting

MailXstream Does Not Appear in the Zapier App Search

Cause: MailXstream is a private Zapier app and has not been added to this Zapier account yet.

Solution: Open **Support** in MailXstream, click **Add MailXstream to Zapier**, and accept the invitation while signed in to the Zapier account you build Zaps with. See [Adding the MailXstream App to Zapier](#).

"Order name is already used"

Cause: Order names must be unique, and the Zap sent one that already exists — usually a typed constant.

Solution: Clear the **Order Name** field to use `auto`, or map something that varies per run.

The Address Arrived Wrong or Empty

Cause: Almost always the mapping, not MailXstream. A lookup missed, or fields were placed on one line.

Solution: Open the run in **Zap History** and read the Send Document step's **Data Out**. If the address is empty, the lookup found nothing — add a **Filter** step so those runs stop instead of submitting. If the lines are run together, re-open the field and confirm each mapped value is on its own row.

The Template's Fields Are Missing

Cause: No template selected yet, or the connection cannot reach that template.

Solution: Select the **Template** first — the fields below appear only afterwards. If the dropdown is empty, confirm your account has template access, and that the environment on your connection matches where the template lives.

"Your MailXstream access token was rejected"

Cause: The token was deleted, expired, or belongs to the other environment.

Solution: Create a new token on the environment your connection uses, then reconnect the account in Zapier.

"Your MailXstream connection predates environment selection"

Cause: The connection was made before environments existed.

Solution: Reconnect the account and choose Production or Test. Existing Zaps keep their settings.

A Trigger Did Not Fire

Cause: Polling delay, the order not yet meeting the condition, or the Zap being off.

Solution: Confirm the Zap is on, check **Zap History** for polls, and confirm in MailXstream that the order actually reached the state. Remember Order Completed fires only at full completion, not at each stage.

"Could not download the file from the previous step"

Cause: The file from the earlier step was unavailable when MailXstream tried to fetch it.

Solution: Re-test the earlier step to refresh it. If it persists, confirm that step still produces a file and that its app connection is healthy.

The Order Went to the Wrong Environment

Cause: The MailXstream step used the wrong connection.

Solution: Check the connection name on the step — it ends with (Production) or (Test) . If it ends with (Unknown) , reconnect it.

Frequently Asked Questions

Can I use one Zap for several templates? No. The template is chosen at setup time. Use one Zap per template, or a **Paths by Zapier** step to route to different MailXstream actions.

Can I submit more than one document in an order? The Send Document action takes one document per order. For several documents, either trigger the Zap once per document or combine them into one PDF beforehand.

How do I stop a Zap that is submitting wrongly? Turn the Zap off in Zapier — that stops new orders immediately. Orders already submitted must be cancelled in MailXstream, and only before processing gets too far.

Does a Zapier-submitted order look different? Only in its source, recorded as `Zapier - 2.0.0`. Processing, pricing, and tracking are identical.

Can I mark orders as tests from Zapier? Test orders are controlled by the template, not by the Zap. Use a Test-environment connection to avoid real mail while building.

Do I need a paid Zapier plan? Any multi-step Zap — anything with a lookup, filter, or formatter — needs a paid plan. Only the simplest two-step Zaps fit the free tier.

What happens if MailXstream is unavailable? Zapier retries automatically and shows the run as errored in Zap History. You can replay it once service returns.

Glossary

Term	Definition
Action	What a Zap does after it triggers
Address File	Uploaded recipient list for a mass mailing
Connection	Stored link between Zapier and one MailXstream account and environment
Delivery Address	Where a mail piece is sent, one line per row
Environment	Production (real mail) or Test (sandbox)
Mapping	Inserting a value from an earlier step into a field
Order	One submission to MailXstream
Polling	Zapier checking on a schedule for new events
Task	One action run by a Zap; counts against your Zapier plan
Template	The MailXstream configuration defining paper, envelope, and mailing options
Trigger	The event that starts a Zap
Zap	An automated workflow: one trigger plus one or more actions

Support

Contact Methods

Method	Detail	Best for
Email	support@mailxstream.com	Template, order, and account questions
Phone	1-877-365-MAIL (6245)	Urgent problems with live mail
Zapier Support	https://zapier.com/app/get-help	Zapier platform, plans, and task limits

Which to Contact

Symptom	Contact
Order rejected, or a template question	MailXstream support
An order printed or mailed incorrectly	MailXstream support
A Zap not triggering, or task limits	Zapier support
Mapping and multi-step Zap questions	Zapier support

Before Contacting Support

Have ready:

- The **Zap History** entry, which shows what was sent and what came back
- The **Order ID** or Order Name, if one was created
- The **template name**
- Whether the connection is Production or Test

Changelog

Version 1.0 — August 2026

- Initial release of the Zapier Integration Guide
- Covers Zapier app 2.0.0 (per-connection Production/Test environment selection)
- Detailed instructions for delivery addresses and mass mailing address files
- Nine end-to-end scenarios

MailXstream — Business-Critical Print-to-Mail

MailXstream is a registered trademark of eVanguard Solutions, Inc. Zapier is a registered trademark of Zapier, Inc.

<https://mailxstream.com> | 1-877-365-MAIL (6245) | support@mailxstream.com