

Forward an Office Inbox Into Pipeline

Last Modified on 09/04/2026 11:08 am EDT

Point *listings@* or *contracts@* at a maildrop address and everything sent there uploads itself. Gmail asks for a verification code first — and we're the ones who can read it.

Introduction

Most offices have an inbox that documents arrive at: *listings@*, *contracts@*, *closings@*. Someone opens it, saves the attachments, and uploads them. That's a job that shouldn't exist, because an auto-forward rule can hand every one of those messages straight to Pipeline.

Setting the rule takes two minutes in Gmail or Office 365. The part that stops people is what Gmail does next: it sends a verification code to the address you're forwarding to, and asks you to type it back in. That address is a maildrop, and a maildrop isn't an inbox: you can't open it and read the code.

Which is why this one includes us. We can pull the code out of the maildrop's email log and send it to you.

When to Use This

- Documents arrive at a shared office address and someone re-uploads them by hand every day.
- You want *listings@* or *contracts@* to feed Pipeline automatically.
- Gmail is asking for a confirmation code sent to an address you can't open.
- You set up forwarding, it says pending, and nothing has arrived.
- A vendor or lender sends documents to one address and you want them filed without a middle step.

1. Get the Address You're Forwarding To

Decide which user's Unassigned maildrop address the office inbox should feed, and copy it. If the forwarded mail should be credited to a shared identity rather than a person, this is the same decision as setting up a scanner — a dedicated user is often the cleaner answer.

Email your documents to a maildrop address and Pipeline pulls the attachments in — no logging in, no uploading.

See [Maildrop](#)

2. Set the Forwarding Rule

In Gmail, add the maildrop address as a forwarding address and choose to forward incoming mail to it. In Office 365, create a rule on that mailbox that forwards messages to the maildrop address. Either way it's the ordinary auto-forward feature — there's nothing Pipeline-specific to configure.

Forward everything, not just attachments. A message with no attachment produces nothing on our end, so a broad rule costs you nothing and a narrow one risks missing documents sent in unusual ways.

3. Get the Gmail Verification Code From Us

Gmail won't activate the forward until you enter a code it emails to the destination — and the destination is a maildrop, which discards message bodies once it has pulled the attachments out. There's no self-serve way to read it.

Email us and we'll dig it out of the email log for that address. Send two things and it's a single round trip: the maildrop address you're forwarding to, and roughly when Gmail sent the verification. Then paste the code into Gmail and the forward goes live.

Need a hand? The quickest way to reach us is right inside Pipeline — and a real person on our team reads every message.

See [Get Help](#)

Office 365 doesn't require this — the code is a Gmail requirement. If you're on Microsoft, your rule is live as soon as you save it.

4. Confirm What's Arriving

A forwarding rule is invisible once it's running, so give yourself a receipt. With acknowledgements on, every message that arrives at the address produces a reply saying what uploaded and what didn't — which tells you the forward is working, and tells you the day it stops.

Email docs into Pipeline and get a receipt back telling you what uploaded, what didn't, and why.

Ask Pipeline to write back every time you email docs in, and you'll know what uploaded and what didn't without going to look.

See [Email Acknowledgement](#)

5. File What Lands in the Queue

Forwarded documents arrive in Unassigned like anything else emailed in, carrying the original sender and subject line — which is usually enough to work out which deal they belong to without opening them.

A doc has landed in Unassigned and now it needs a home. Assigning is the step that puts it on the right transaction, with the right name and the right permission category.

See [Doc Assignment](#)
