email at *awesome*

Verify every email before it enters your workflow.

Emails bounce, people change jobs, domains go stale, catch-alls hide, role-based emails, fake addresses, and the occasional typo that ruin your sender rep.

Email at Awesome runs every email through syntax checks, MX lookups, SMTP handshakes, catch-all probes, and disposable-domain checks in real time, so you only send to addresses that actually exist. Bad addresses never reach your inbox, and your sender reputation stays clean.

[REAL-TIME SMTP CHECKS]

Beyond syntax checks, we connect to the recipient's mail server and verify the address exists before you hit send. Catch typos, dead domains, and mailboxes that don't exist before they bounce.

[CATCH-ALL DETECTION]

Catch-all domains accept everything, so you can't tell what's real from what's fake. We flag catch-all addresses separately so you know which ones need extra scrutiny before you send.

[DISPOSABLE EMAIL FILTERING]

Block throwaway addresses from Mailinator, Yopmail, and thousands more. Keep your lists full of people who actually want your emails, not people who don't want to be bothered.

[ROLE-BASED ADDRESS DETECTION]

Flag role-based addresses like info@, support@, and admin@ so you can decide whether to include them in your campaigns or skip them entirely.

[BULK VALIDATION]

Upload a list and we'll process it at scale. Get back a clean, deliverable list with every invalid, risky, and disposable address flagged and ready to remove.

[API INTEGRATION]

Plug into our API to validate addresses at the point of signup, in your CRM, or anywhere else in your stack. Bad addresses never get in in the first place.

[DOMAIN HEALTH CHECKS]

Check whether a domain has valid MX records before you waste time on it. Spot dead domains and misconfigured mail servers before they affect your sender reputation.

[DELIVERABILITY SCORING]

Every address gets a score based on syntax, domain, SMTP response, and catch-all status. Sort your list by confidence and focus your sends on the addresses most likely to land.

[DOCUMENTATION]

Full API docs with examples in every major language. Get your first validation running in minutes, not days.

[PRICING]

Pay as you go starting at $0.003 per email. No subscriptions, no minimums, no expiring credits. Bulk discounts kick in automatically as you scale.

[SOCIAL PROOF]

Trusted by 2,847 teams. Average bounce rate dropped from 4.2% to 0.3% within the first month. Average time to first value: 3 minutes.

[FAQ]

Q: Do you store the emails we validate?
A: No. We process each address in real time and don't retain email addresses after the validation response is returned.

Q: What happens with catch-all domains?
A: We flag them as catch-all so you can decide how to handle them. Some people include them, others skip them. We give you the data to make that call.

Q: How accurate is SMTP verification?
A: We achieve 98.7% accuracy on our benchmark dataset. Some mail servers rate-limit or block probes, we handle retries and fallbacks automatically.

Q: Can I integrate with my CRM?
A: Yes. We have webhooks and a REST API that plug into any workflow. Most customers integrate with their CRM in under an hour.

Q: What's your uptime?
A: 99.9% over the last 12 months. Status page is at status.email-at-awesome.com.

[CTA]

Start verifying for free. 500 validations per month, no credit card required.

[FOOTER]

email at awesome · Made with care · Privacy · Terms · Status · Contact

---

INTRODUCTION

Who this guide is for

Ops and growth teams at SaaS, D2C and fintech companies, from 10-person startups to 200-person companies, who run lead gen, trial or demo flows, and need to know which of their email addresses actually work before they hand the list to sales or start a campaign.

If your bounce rate is creeping up, your sender reputation is slipping, or your sales team keeps chasing dead ends, this guide is for you. It covers what to check, what to skip, and how to build a workflow that keeps your lists clean without slowing your team down.

CONTENTS

What this guide covers

[SECTION · THE FORMAT]
Marcus's 30-minute audit. The frame he's still using. Six checks, one sample, one decision. Why most teams never run it twice, and what it costs them when they don't.

[SECTION · THE CHECKS]
The four checks that matter: syntax, MX, SMTP, catch-all. What each one catches, what it misses, and why you need all four in order.

[SECTION · THE DECISIONS]
Hard bounces, catch-alls, and the gray zone. How to decide what to do with each result before you start sending.

[SECTION · THE SAMPLE]
Why 200 addresses is enough. How to pick the sample so it represents the whole list. What to do with the results once you have them.

[SECTION · WHAT'S NEXT]
Turning a one-time audit into a habit. Where to run the checks in your stack. Which tools to use at each step.

The hidden cost of bad email data

Marcus's team at a Series B fintech ran a lead gen campaign last quarter. 4,200 addresses, purchased from a list broker, looked clean on the surface. Roughly 18% bounced within the first hour. The sales team spent the next two weeks chasing dead ends while their sender reputation quietly dropped. By the time they noticed, the damage was done.

Wasted sends	4,200
Hard bounces	760
Sales hours lost	~80
Sender rep recovery	6 weeks

The list broker's guarantee covered the refund. It didn't cover the pipeline, the time, or the trust they burned with their own sales team.

This kind of thing happens quietly. Nobody tracks how much time their team spends chasing bad leads. Nobody calculates what a bounce does to their sender reputation over a quarter. The cost is real, but it's spread across so many small moments that it never shows up as a single line item.

What this guide covers

The rest of this guide walks through a simple audit that any ops or growth team can run in an afternoon. It covers four checks, what each one catches, and what to do with the results. It doesn't require a data science degree or a new tool. The goal is to give you a clean list and a repeatable process so you can do it again when it matters.

Why list cleaning happens too late

The standard SMB approach to email data quality is to clean the list before a campaign. Import the new leads, run them through a verification tool, remove the bounces, and send. This works once. The problem is that it's the right thing to do, but it's not a habit. It gets skipped when the team is busy, when the deadline is tight, when someone thinks the list looks fine.

The asymmetry

The cost of a bad email address happens downstream, not at the point of collection. Bounces show up days later, sender reputation drops over weeks, and by the time you notice, the damage is done. The time to clean a list is the right thing to do, but it's also the time when everyone is most tempted to skip it because the campaign is already late.

Decay is real

Even a list that was clean when it was bought, and that passed the checks at import, ages within a few months. People change jobs, domains expire, mailboxes get deactivated. A list that was 95% accurate at purchase can easily drop below 80% accurate six months later. This is why one-time cleaning isn't enough. The only fix is a recurring check, or at least a check before each major campaign.

The Email at Awesome API in plain English

An API is a way for one system to talk to another. Instead of a human typing addresses into a form, your CRM sends them to a server, the server checks them, and the answer comes back in a few seconds. Email at Awesome exposes a single endpoint for this. You send it an email address, it runs the checks, and it returns a structured response with the result, the reason, and a confidence score.

What gets sent, and what comes back

A typical request sends one email address or a batch of up to 500 at once. The response includes the syntax result, the MX record, the SMTP response, the catch-all flag, and a deliverability score that combines all of these signals into one number.

SYNTAX	MX RECORD	SMTP RESPONSE	CATCH-ALL
valid	found	250 OK	not detected

The catch-all risk score

Catch-all domains accept mail for any mailbox, so they look deliverable on paper but may silently bounce later. The API flags catch-alls separately and gives them a lower score than confirmed-valid addresses. This lets you decide your own threshold: some teams treat catch-alls as safe enough, others want to skip them entirely. Both are reasonable depending on your sender reputation and how much you rely on each lead.

Rate limits and batch processing

Single requests are fast, but for large lists you'll want to use the batch endpoint. The API accepts up to 500 addresses per request and returns results in the same order you sent them. Rate limits are generous for normal use and scale with your plan. Bulk imports can run overnight if you're cleaning a large list from a CRM export.

Five places to embed verification in your stack

Email at Awesome exposes a single API endpoint. You can call it from any tool that can make a web request. Zapier, Make, n8n, Google Sheets, or your own backend. The right place depends on where the bad data first enters your system and how much friction you can tolerate at that point.

Forms

Validate at the moment someone fills out a form. Catch typos and disposable addresses before they ever reach your CRM. Adds a few seconds to the form but prevents the problem from spreading.

Webhooks

Run a check every time a new lead arrives from a form, a list import, or a third-party source. Flag or quarantine bad addresses before your sales team sees them.

Scheduled batch

Run a weekly or monthly sweep over your full contact database. Catch decay that happened since the last check. Useful for teams that don't import new leads every day but still want their list to stay clean.

Pre-campaign

Check the list you're about to send to. Last chance to remove bounces before they hit your sender reputation. Pairs well with a weekly batch run so the list you send from is already mostly clean.

Manual

Paste a few addresses into a spreadsheet or a form when you're unsure about one specific lead. Useful for edge cases and for teaching your team what the tool does before they automate it.

The four workflow patterns

Once the API is plumbed in, your next question is how to decide what to do with it. Four patterns cover most teams. Pick the one that matches your risk tolerance and how much manual review your team can handle.

Verify and block	Only send to valid addresses. Safest option for protecting sender reputation.
Verify and review	Flag risky results for a human to look at before deciding. Slower but catches edge cases.
Verify and score	Attach the score to the lead record and let downstream tools decide how to weight it. Most flexible.
Verify and clean	Run a weekly sweep over the full database to remove decay that happened since the last check.

The four-week rollout

Email at Awesome offers a 14-day free trial. This rollout schedule fits within that trial and gives you enough data to make a real decision about whether the tool is worth keeping.

Week 1	Run the audit on a sample of 200 addresses. Compare results to your last campaign. Decide if the tool is worth a deeper look.
Week 2	Add a single form or webhook. Watch the flagged addresses for two weeks. Check how many your team would have missed.
Week 3	Run a full list through the batch endpoint. Review the results with someone who owns the CRM. Agree on what to do with each result.
Week 4	Decide which workflow pattern fits your team. Set up the recurring check. Document the decision so it doesn't get relitigated later.

The engineering brief template

This is the minimal ticket someone can pick up. Copy it into your tracker, fill in the blanks, and hand it to the person who owns your CRM integration. Total implementation time for a simple setup is usually under an hour.

Goal	Add email verification to [form / webhook / import] using Email at Awesome API.
Scope	[specific form, webhook endpoint, or import job] only. Do not touch other flows in this ticket.
Acceptance	Invalid emails rejected or flagged. Valid emails unchanged. Logs show API response for each call.
Rollout	Ship behind a feature flag. Run for 2 weeks in parallel with the old flow. Compare flagged rates before removing the old path.
Owner	[name]

Picking your Email at Awesome plan tier

Volume is the main decision here. Most teams start on the free tier and move up once they hit the limit or need a feature they didn't expect. Prices below are illustrative; check the current pricing page before you commit.

TIER	VOLUME / MO	KEY FEATURE
Free	500	Single checks, no credit card
Starter	2,000	Bulk upload, CSV export
Pro	10,000	Webhooks, API access, priority support
Pro+	50,000	Dedicated IP, custom rate limits
Pro+	100,000	Dedicated IP, custom rate limits

Start verifying at the source this week.

Pick one form or one webhook. Run it for two weeks. Compare the flagged rate to what your team would have missed. Decide if the tool is worth a deeper look. Most teams do it for a single form first, learn what the numbers look like, and then roll it out to the rest of the stack once they trust the output.

The hard part isn't the technical work. The hard part is remembering to check the list before the campaign goes out instead of after the bounces come back. Email at Awesome automates the check so it stops being something your team has to remember to do.

Documentation lives at email-at-awesome.com/docs. Start with the free tier and see how many of your addresses actually work.

APISO

Turn documentation into production-ready API integrations. Import specs, test endpoints, and ship faster.

Get started →