VERTEX FRONTIER

2026 EDITION · FREE DOWNLOAD

The Self-Hosted Email Field Guide

Mailcow on a VPS — DNS, authentication, warm-up, hardening, and recovery — condensed into a printable reference you can keep next to your terminal.

SPF DKIM DMARC POSTFIX RSPAMD DOCKER UFW FAIL2BAN

BROUGHT TO YOU BY
Vertex Frontier — vertexfrontier.com

FIELD GUIDE
No. 01 / Email Infrastructure

CHAPTER 01

Start Here

This field guide is the operational companion to *"How to Build a Self-Hosted Email Server With Mailcow."* It strips out the narrative and keeps only what you need at the terminal — record values, commands, checklists, error decoders, and the warm-up calendar.

Print it, pin it next to your monitor, and tick off each step as you go. Every section maps back to a numbered step in the full article, so if anything is unclear you can jump back to the long-form explanation in seconds.

PRINCIPLE

Setup is 20% of the work. Deliverability is the other 80%. This guide treats both with equal weight — do not skip the warm-up section just because the server boots.

CHECK VPS Prerequisites

Before you touch a terminal, your VPS must satisfy every line below. A single failure here dooms deliverability before you send your first message.

- [ ] Port 25 outbound **open** (not blocked)
- [ ] Custom **PTR record** supported
- [ ] Clean IP — **not** on any blocklist
- [ ] Minimum **6 GB RAM** (8 GB with ClamAV)
- [ ] 20 GB disk minimum
- [ ] Static IPv4 (and IPv6 if you commit to it)
- [ ] A domain whose DNS zone you control

PICK Provider Cheat Sheet

Port 25 and PTR support vary by provider. The wrong choice here is the single most common reason teams abandon a self-hosted email project mid-setup.

Provider	Port 25	PTR Record	Notes
Hetzner	OPEN	SELF-SERVE	Recommended default. Edit PTR in the Networking tab.
Vultr	TICKET	YES	Open a support request to unblock port 25.
DigitalOcean	TICKET	YES	Default blocks 25; some new accounts never get unblocked.
AWS / GCP	BLOCKED	YES	Request exception; rarely granted to fresh accounts.

CHAPTER 02

DNS Records Master Sheet

The single most decisive factor in self-hosted email. Get every record right and the rest of the guide becomes routine. Get one wrong and the server silently fails.

Type	Host	Value	Purpose
A	yourdomain.org	203.0.113.10	Root domain → server IP
A	mail.yourdomain.org	203.0.113.10	Mail subdomain → server IP
MX	@ (root)	mail.yourdomain.org (priority 10)	Routes inbound mail to your server
CNAME	autodiscover	mail.yourdomain.org	Auto-config for Outlook
CNAME	autoconfig	mail.yourdomain.org	Auto-config for Thunderbird
TXT	yourdomain.org	v=spf1 mx a -all	SPF — authorizes senders
TXT	_dmarc.yourdomain.org	v=DMARC1; p=none	DMARC — starts in monitor mode
TXT	dkim._domainkey.yourdomain.org	(public key from mailcow panel)	DKIM — cryptographic signing
AAAA	mail.yourdomain.org	2001:db8::1	IPv6 — only if you commit to IPv6
PTR	(IP address, set by VPS provider)	mail.yourdomain.org	Reverse DNS — set via support ticket

PTR — THE RECORD YOU CANNOT SET YOURSELF

A PTR record maps your server's IP *back* to `mail.yourdomain.org`. Receiving servers use it to confirm the sending IP genuinely belongs to who it claims to be. Only your **VPS provider** controls reverse DNS for their IP ranges — open a support ticket, most handle it within minutes.

THE IPV6 TRAP NOBODY MENTIONS

If your VPS has IPv6 enabled and you skip the AAAA record and IPv6 PTR, some receivers will route over IPv6 first, find no matching reverse DNS, and reject the message outright — even though your IPv4 setup is flawless. Two honest options: do it properly (AAAA + IPv6 PTR), or disable IPv6 for mail entirely.

```
# /opt/mailcow-dockerized/data/conf/postfix/extra.cf# Force IPv4-only sending if your IPv6 setup is inconsistent.inet_protocols = ipv4
```

VERIFY PROPAGATION

Use `mxtoolbox.com` — search for `mail.yourdomain.org` and confirm the A record resolves. Propagation can technically take 24 hours, though it is usually much faster.

CHAPTER 03

The Authentication Trio

SPF says who can send. DKIM proves the message was not tampered with. DMARC decides what happens when either fails. Skip one and the other two lose most of their value.

SPF Sender Policy Framework

A DNS TXT record listing which mail servers are allowed to send on behalf of your domain. For a single-server setup, the value below is sufficient.

```
v=spf1 mx a -all
```

-all means "reject anything not explicitly listed" — the strictest, most trustworthy setting.

DKIM DomainKeys Identified Mail

Attaches a cryptographic signature to every outgoing message. Receiving servers verify the signature against your public key in DNS. Mailcow generates this key automatically — copy the TXT value it shows you into a record named `dkim._domainkey.yourdomain.org`.

COMMON MISTAKE

Creating the DMARC record before DKIM has fully propagated will cause your own legitimate mail to fail authentication.

DMARC Policy That Ties It Together

Tells receiving servers what to do when SPF or DKIM checks fail, and sends you reports on your domain's email activity. Start conservative at `p=none` and tighten over four weeks.

```
v=DMARC1; p=none
```

Record name: `_dmarc.yourdomain.org`

WAIT 48 HOURS BEFORE TIGHTENING

Give DNS and your DKIM signature at least 48 hours to propagate cleanly before moving DMARC past `p=none`. Jumping straight to `p=reject` on a fresh server is one of the fastest ways to have legitimate mail silently dropped.

TLSA Encrypting the Connection

Ties your mail server's TLS certificate to DNS, stopping man-in-the-middle attacks against encrypted mail connections. Mailcow generates the value for you, but TLSA only works if you also enable **DNSSEC** on your domain — usually a single toggle in your registrar's settings.

INDUSTRY CONTEXT

As of February 2024, Gmail's official sender guidelines require senders of 5,000+ messages a day to authenticate their outgoing email. Enforcement tightened further in November 2025, with Gmail ramping rejections of non-compliant traffic. Skipping this trio risks outright rejection, not just spam-folder placement.

CHAPTER 04

Command Library

Every command from the full article, organized by step. Run them in order. Never work as root beyond step 1.

1 SSH in, update the system, install dependencies.

```
ssh root@yourdomain.org
apt update && apt upgrade -y
apt install git curl openssl -y
```

2 Create a non-root user with sudo privileges.

```
adduser tony
usermod -aG sudo tony
su - tony
```

3 Add a swap safety net (cheap insurance against OOM kills during ClamAV signature updates).

```
sudo fallocate -l 4G /swapfile
sudo chmod 600 /swapfile
sudo mkswap /swapfile
sudo swapon /swapfile
echo '/swapfile none swap sw 0 0' | sudo tee -a /etc/fstab
```

Confirm with `free -h`.

4 Install Docker and reboot.

```
curl -fsSL https://get.docker.com -o get-docker.sh
sudo sh get-docker.sh
sudo usermod -aG docker tony
sudo reboot
```

5 Clone mailcow and run the config generator.

```
umask 0022
cd /opt
sudo git clone https://github.com/mailcow/mailcow-dockerized
cd mailcow-dockerized
timedatectl list-timezones | grep York
sudo ./generate-config.sh
```

Answer: hostname `mail.yourdomain.org`, time zone, branch = `stable`.

6 Pull and start the stack.

```
sudo docker compose pull
sudo docker compose up -d
sudo docker compose ps
```

Every container must show status **Up**. If anything shows "unhealthy," inspect logs: `sudo docker compose logs <container-name>`

7 Open `https://mail.yourdomain.org`, log in as **admin** / **moohoo**, change the password immediately, then add your domain and first mailbox.

Hostname vs. address: `mail.yourdomain.org` is the server; `tony@yourdomain.org` is the address. Do not mix these up.

> **POSTFIX IPV4-ONLY (OPTIONAL)**
>
> If your VPS provider's IPv6 support is inconsistent, edit `/opt/mailcow-dockerized/data/conf/postfix/extra.cf` and add `inet_protocols = ipv4`, then restart Postfix.

CHAPTER 05

IP Warm-Up Calendar

A technically perfect setup can still fail — deliverability is about reputation, and reputation is earned over time. A brand-new IP has no history. Send a burst of 100 emails on day one and anti-spam systems read it as bot behavior, not enthusiasm.

WEEK 1 · INCUBATION

DMARC: `p=none`

Send 5–10 *manual* emails a day to real people who will open and reply. Zero automated or bulk mail. No marketing blasts.

WEEKS 2-3 · TRUST BUILDING

DMARC: `p=quarantine`

Review your DMARC reports. If SPF and DKIM are passing cleanly, raise volume to 20–50 emails a day. Continue engaging recipients.

WEEK 4+ · ENFORCEMENT

DMARC: `p=reject`

Your domain is now protected against spoofing and your IP has a genuine track record with major providers. Scale volume sensibly.

CHECK Pre-Send Deliverability

Do not email a real person until every box below is checked. Use `mail-tester.com` to score your setup out of 10. Aim for 9+.

- [ ] IP is not on a spam blocklist (`mxtoolbox.com`)
- [ ] PTR record maps your IP to `mail.yourdomain.org`
- [ ] MX record points to `mail.yourdomain.org`
- [ ] DKIM shows valid in the mailcow panel
- [ ] SPF set: `v=spf1 mx a -all`
- [ ] DMARC set with `p=none` to start
- [ ] DNSSEC and TLSA both active
- [ ] Every mailcow container shows status "Up"
- [ ] mail-tester.com score is 9/10 or higher

CREATE A DMARC MAILBOX

Set up `dmarc@yourdomain.org` to catch the automated reports providers send you. Read them. They surface a broken record long before it becomes a full-blown deliverability crisis.

THE ROLLOUT MISTAKE ALMOST NOBODY WARNS YOU ABOUT

Rushing the warm-up undoes everything the DNS work accomplished. Deliverability is a trust score you build over weeks, not a switch you flip once. Send a burst of a hundred emails from a fresh IP on day one and you will get blocklisted before you have had a real conversation.

CHAPTER 06

Reading the Error Messages

A bounce code tells you something failed. Rspamd's History tab tells you what to actually fix. Always check both before assuming your whole setup is broken.

POSTFIX SMTP Error Codes

Code	Type	Meaning
421	TEMP	Receiving server busy or rate-limiting you. Will retry automatically.
450 / 451	TEMP	Greylisting, or a local mailbox issue on the recipient's side.
550	PERM	Rejected — usually a blocklisted IP, missing PTR, or failed SPF/DKIM.
551 / 553	PERM	Mailbox doesn't exist, or "From" address doesn't match an authorized domain.
554	PERM	Transaction failed — often a catch-all for "flagged as spam."

RSPAMD Symbol Decoder

Symbol	What It Means & How to Fix
DKIM_INVALID	DKIM signature failed verification. Usually DNS not propagated yet.
DKIM_REJECT	DKIM hard-failed. Re-check `dkim._domainkey` record.
DMARC_POLICY_REJECT	Message failed your own DMARC policy. Often during the 48-hour propagation window.
RBL_SPAMHAUS	Your sending IP is listed on a blocklist Rspamd checks. Investigate immediately.
MISSING_MID	Message is missing a Message-ID header. Hallmark of poorly formed automated mail.
RCVD_IN_DNSWL	Positive signal — your IP is on a trusted allowlist. Keep up the good behavior.

LIVE Reading the Logs

A bounce message tells the sender something failed. The logs tell *you* why. Always inspect both before assuming the server is broken.

```
# Last 100 Postfix log lines
sudo docker compose logs postfix-mailcow --tail=100

# Fail2ban status (currently banned IPs)
sudo docker compose exec netfilter-mailcow fail2ban-client status

# Rspamd fuzzy learning statistics
sudo docker compose exec rspamd-mailcow rspamadm control fuzzy_stat
```

SMARTHOST FALLBACK

If Microsoft 365 / Exchange Online keeps rejecting mail from a brand-new IP regardless of correct configuration, route outbound through a smarthost (Amazon SES, SendGrid, Mailgun) for the first 3–6

months. In mailcow: Configuration → Routing → Relayhost. Switch back to direct sending once your IP reputation is established.

CHAPTER 07

Security, Backup & Monitoring

A mail server on a public IP gets probed constantly. A compromised mail server can be weaponized to spam the entire internet under your domain's name. These three checklists are not optional.

HARDEN Server & SSH

- [ ] Rotate every default password (mailcow admin + every mailbox)
- [ ] UFW open only: `22, 25, 80, 443, 465, 587, 993, 995`
- [ ] SSH key-based authentication only; disable password login
- [ ] Disable root login over SSH
- [ ] Enable 2FA on mailcow admin (TOTP, not SMS)
- [ ] Enable 2FA on every SOGo mailbox that matters
- [ ] Save recovery codes *off* the server

```
# Update mailcow to latest security patches
cd /opt/mailcow-dockerized
sudo ./update.sh
```

BACKUP Automated & Off-Site

An unautomated backup is a backup you will forget to run right before you need it. A backup on the same disk as the source is a copy that fails at the same time as the original. Automate, then push off-server the same day.

```
# Manual one-off backup
cd /opt/mailcow-dockerized
sudo helper-scripts/backup_and_restore.sh backup all

# Nightly automated backup + retain 7 days
crontab -e
# add:
0 3 * * * cd /opt/mailcow-dockerized && sudo helper-
scripts/backup_and_restore.sh backup all --delete-days 7
```

PUSH OFF-SITE

Add `rclone` or `restic` to the same cron job and push to AWS S3 or Backblaze B2. Ten extra minutes of setup is the difference between "annoying afternoon" and "the business lost a week of email permanently."

MONITOR Continuous Health

- [ ] Blocklist monitoring: **HetrixTools** or **MXToolbox Monitors** (free)
- [ ] Configure push notifications (email or Telegram bot)
- [ ] Review DMARC reports on a schedule, not only when something feels wrong
- [ ] Weekly: `sudo docker compose ps` even when nothing seems wrong
- [ ] Tighten Fail2ban thresholds in System → Configuration → Fail2ban Parameters
- [ ] Mark misclassified messages in Rspamd's History tab to keep fuzzy learning accurate

V VERTEX FRONTIER

YOU'RE READY.

Ship it. Then keep it patched.

The DNS work is tedious. The warm-up demands patience. SSH hardening is not optional. But the first time a test email lands cleanly in a Gmail inbox — from your own domain, with nobody else's name on the infrastructure — it's worth every hour.

Read the Full Guide →
The long-form walkthrough with case studies, cost calculator, and FAQ.

Get the Next Field Guide →
Subscribe to be notified when the next printable reference drops.

Join the Conversation →
Follow Vertex Frontier for new guides on self-hosting, privacy, and infrastructure.