

DKIM Settings

DKIM Settings

Admin path: **Content Checks > DKIM Settings** (`view_dkim_settings.cfm`, `inc/get_dkim_settings.cfm`, `inc/dkim_save_settings.cfm`, `inc/dkim_set_settings.cfm`, `inc/dkim_generate_config_file.cfm`, `inc/dkim_generate_keytable.cfm`, `inc/dkim_generate_signingtable.cfm`, `inc/dkim_generate_hosts.cfm`, `inc/dkim_generate_domains.cfm`, `inc/restart_opendkim.cfm`, `inc/generate_postfix_configuration.cfm`).

This page controls **inbound DKIM verification** and the **OpenDKIM runtime configuration** that also drives outbound signing. DKIM ([RFC 6376](#)) lets a sending domain attach a cryptographic signature (`DKIM-Signature: v=1; a=rsa-sha256; d=example.com; s=mail1; ...`) covering selected headers and a hash of the message body; receivers fetch the public key at `<selector>._domainkey.<domain>` in DNS and verify the signature. Unlike SPF, DKIM survives most forwarding — the signature stays attached to the message and verifies wherever the body and signed headers remain unchanged.

Per-domain key generation (selector, RSA 1024 / 2048, DNS TXT record to publish) is managed elsewhere — on the [Email Server Domains](#) page via `edit_domain_dkim.cfm`, which writes rows into the `dkim_sign` table. This Settings page configures the OpenDKIM daemon's runtime behavior and maintains the verification-side bypass lists.

Two OpenDKIM instances, one config page

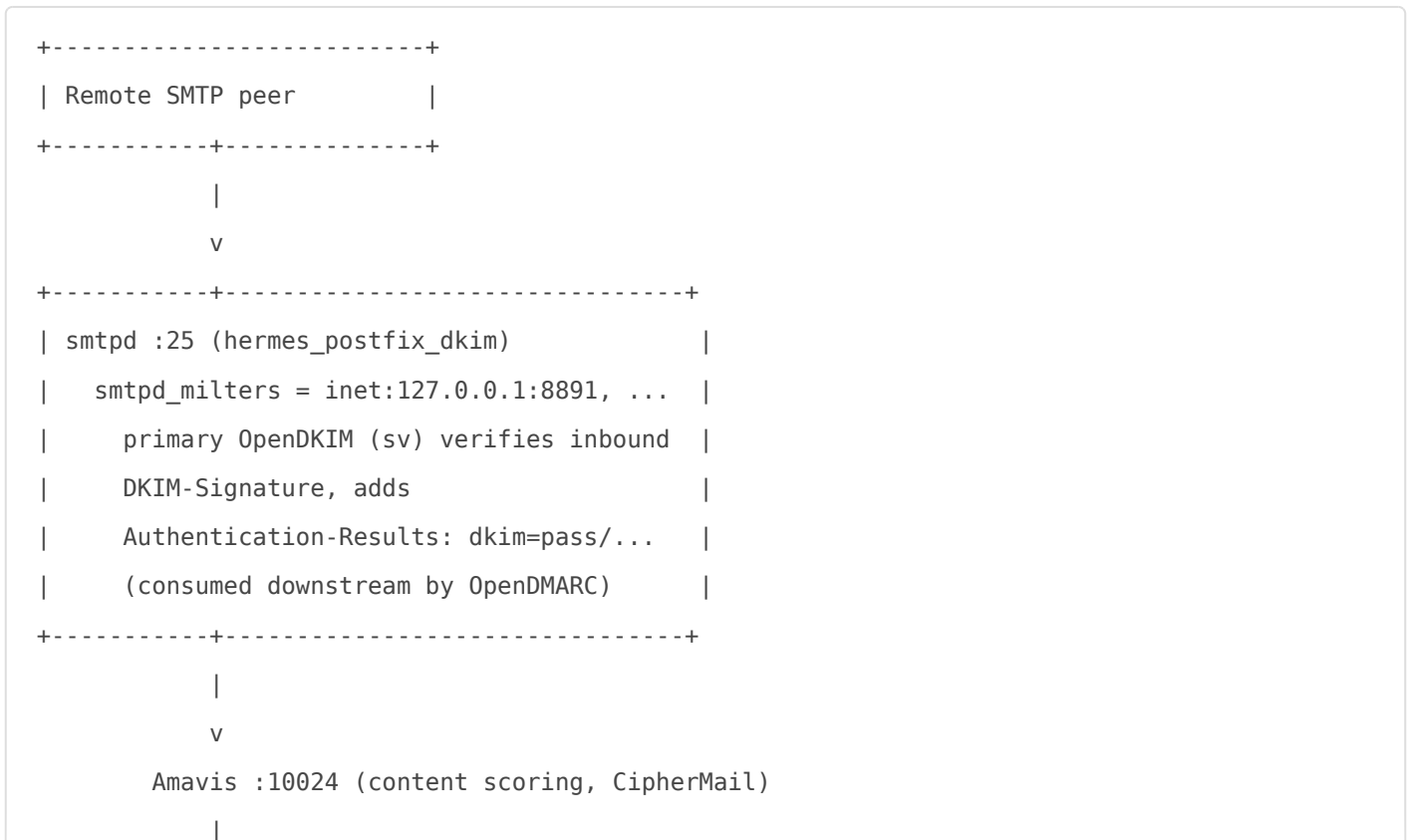
To avoid the body-modification trap that breaks any signer running after a body-modifying milter, Hermes (issue #232) runs **two separate OpenDKIM instances** inside `hermes_postfix_dkim`:

Instance	Config	Socket	Mode	Role
----------	--------	--------	------	------

Primary	<code>/etc/openskim.conf</code>	<code>inet:8891@0.0.0.0</code>	<code>sv</code> (sign + verify)	Verifies inbound DKIM at <code>smtpd :25</code> ; signs outbound at <code>:587</code> / <code>:465</code> (submission ports — pre-Amavis, pre-CipherMail)
Sign-only	<code>/etc/openskim-sign.conf</code>	<code>inet:8892@127.0.0.1</code>	<code>s</code> (sign only)	Signs at the <code>:10026</code> re-injection port after Amavis, CipherMail, and the body milter have finished modifying the body. Never adds an <code>Authentication-Results</code> header

Both instances share the same key tables (`/opt/hermes/dkim/KeyTable`, `/opt/hermes/dkim/SigningTable`) and the same trusted-hosts / exempt-domains lists; the page below feeds both. The reason for two instances: a single `sv`-mode OpenDKIM on `:10026` would verify the post-modification body of inbound mail flowing through the re-inject port and emit a spurious `dkim=fail` `Authentication-Results` header. Sign-only mode at `:10026` produces the final outbound signature over the byte sequence the receiver will actually see.

Where DKIM sits in the flow



```

v (reinject)
+-----+-----+
| smtpd :10026 (post-content, post-body-mod) |
|   smtpd_milters = inet:127.0.0.1:8891     |
|     sign-only OpenDKIM at :8892 actually  |
|     signs the final outbound body         |
|     (KeyTable selects per-domain key by   |
|     "*@<domain>" SigningTable match)     |
+-----+-----+
|
| v
| OpenARC seal (if enabled)
|
| v
| Outbound to receiver

```

The actual signing decision happens against the `SigningTable`:

```

# /opt/hermes/dkim/SigningTable
*@example.com      mail1._domainkey.example.com
*@partner.org      k2024._domainkey.partner.org

```

...joined to the `KeyTable`:

```

# /opt/hermes/dkim/KeyTable
mail1._domainkey.example.com
example.com:mail1:/opt/hermes/dkim/keys/mail1_example.com.dkim.private
k2024._domainkey.partner.org
partner.org:k2024:/opt/hermes/dkim/keys/k2024_partner.org.dkim.private

```

Both files are regenerated from the `dkim_sign` table on every key add / enable / disable / delete on the per-domain page.

The two cards on the page

1. DKIM Settings (master toggle + OpenDKIM runtime controls)

DKIM Enabled flips the child row in `parameters` whose `parameter` matches `inet:%:8891` under the `smtpd_milters` parent (and the same under `non_smtpd_milters`). Disabling DKIM here also disables DMARC, mirroring the SPF-disable behavior — DMARC needs at least one of the two to align against. The in-page callout warns about this dependency.

When enabled, nine controls are written to `parameters2` rows in the `dkim` module, then substituted into the OpenDKIM template at `/opt/hermes/conf_files/opendkim.conf.HERMES`:

Control	OpenDKIM directive	Effect
Body Canonicalization	<code>Canonicalization</code> (body half)	<code>relaxed</code> (recommended) ignores trailing whitespace and end-of-line changes; <code>simple</code> requires byte-exact body. Most relays touch line endings, so <code>relaxed</code> is the only practical choice unless you fully control every downstream hop
Headers Canonicalization	<code>Canonicalization</code> (header half)	<code>relaxed</code> lowercases header names and folds whitespace; <code>simple</code> requires headers unchanged. Same reasoning — <code>relaxed</code> survives normal relay reformatting
Default Message Action	<code>On-Default</code>	Catch-all for verification outcomes not covered by the more specific actions below. <code>accept</code> is the recommended default
Bad Signature Action	<code>On-BadSignature</code>	Signature present, present-and-valid in syntax, but verification fails (body or signed-header bytes changed). <code>accept</code> (recommended) lets DMARC + spam scoring make the call
DNS Error Action	<code>On-DNSErrors</code>	The selector's <code>_domainkey</code> TXT record is unreachable or returned SERVFAIL. <code>accept</code> (recommended) — DNS instability is the sender's problem, not yours; do not block real mail on transient resolver failures
Internal Error Action	<code>On-InternalError</code>	OpenDKIM ran out of resources or hit an unexpected runtime error. <code>accept</code> (recommended) prevents silent mail loss when the verifier itself fails
No Signature Action	<code>On-NoSignature</code>	Message arrived unsigned. Many legitimate senders still don't sign — DMARC enforcement is the correct gate for "must be signed", not this knob. <code>accept</code> (recommended)

Control	OpenDKIM directive	Effect
Security Concern Action	On-Security	Signature references a weak algorithm or unusually short key. <code>accept</code> (recommended) — score downstream rather than reject at the milter
Signature Algorithm	SignatureAlgorithm	<code>rsa-sha256</code> (current standard, recommended) or the deprecated <code>rsa-sha1</code> . Many receivers reject <code>rsa-sha1</code> outright; do not change unless you know why

Each "Action" option set is: `accept`, `discard`, `reject`, `tempfail`, `quarantine`. The save handler validates that submitted values are members of this set before writing.

“ **Operational consequence — accept everywhere is intentional.** The recommended baseline accepts on every error and every failure condition because **DKIM at the milter is not a delivery gate**. The verification result is meant to be consumed by DMARC and by spam scoring, not to drop mail. Setting any of these to `reject` means a single sender DNS hiccup or a single intermediate relay rewriting a header can cause real mail to bounce. Leave them at `accept` and let DMARC enforcement (which considers the sender-published policy) make the discard decision.

2. Whitelisted Domains and Trusted Hosts

Two row-per-entry lists that together drive three OpenDKIM directives:

Entry type	OpenDKIM directive(s)	File on disk	Table
Whitelisted Domain	<code>ExemptDomains</code>	<code>/opt/hermes/dkim/ExemptDomains</code>	<code>dkim_bypass</code> (<code>entry</code> , <code>note</code>)
Trusted Host	<code>InternalHosts</code> + <code>ExternalIgnoreList</code>	<code>/opt/hermes/dkim/TrustedHosts</code>	<code>dkim_trusted_hosts</code> (<code>host</code> , <code>note</code>)

Whitelisted Domain exempts the listed sender domain from inbound DKIM verification entirely — OpenDKIM logs the bypass and does not fetch the selector record. Use for known-broken signers whose mail you still need to receive (some legacy mailing-list infrastructure, specific government endpoints with unmaintained selectors).

Trusted Host is dual-purpose. The same entries are written to both `InternalHosts` (mail from these hosts is considered locally originated and will be DKIM-signed on the way out) and `ExternalIgnoreList` (mail from these hosts skips inbound DKIM verification). Accepts IP addresses, CIDR ranges, hostnames, and bare domain names. The Docker subnet (`172.16.32.0/24` by default)

is pre-populated so the post-Amavis re-inject from `127.0.0.1` and the inter-container hops are correctly treated as internal.

The DataTable supports add (textarea — one entry per line, deduplicated), inline edit, single delete, and bulk delete; the row checkboxes carry an `id|type` composite value so the bulk handler can route each delete to the right table.

What this page does NOT control

- **Per-domain DKIM key generation, selector choice, key size, key rotation, and the DNS TXT record to publish.** Those live on the Email Server [Domains](#) page via `edit_domain_dkim.cfm` — one selector / key per domain, stored in the `dkim_sign` table, written under `/opt/hermes/dkim/keys/<selector>_<domain>.dkim.{private,txt}`.
- **The KeyTable and SigningTable content.** These are regenerated from `dkim_sign` rows on every key change; do not edit them by hand.
- **ARC sealing.** The post-modification chain seal is a separate daemon — see [ARC Settings](#).
- **Outbound signing for sub-domains of a signed parent.** OpenDKIM's `*@<domain>` SigningTable match does not implicitly cover `*@sub.<domain>`. If you sign for `example.com` and need `mail.example.com` signed too, generate a separate key for it.

Per-domain key rotation pattern

A working selector-rotation looks like this (operator-side, not a single button on the page):

1. On `edit_domain_dkim.cfm`, generate a new key with a new selector (e.g. existing "mail1" -> new "mail2"). Mark NEW key disabled.
2. Publish the new key's TXT record at `mail2._domainkey.example.com` in authoritative DNS. The old `mail1._domainkey.example.com` record STAYS published.
3. Verify DNS propagation globally.
4. Enable the new key (disables the old one in `dkim_sign` atomically). KeyTable + SigningTable regenerate; OpenDKIM reloads.
5. Outbound mail now signs with mail2; mail signed with mail1 while in flight still verifies because the mail1 TXT record is still live.
6. Wait through the typical re-delivery window (24-72 hours).
7. Delete the old mail1 row in `dkim_sign`; remove the `mail1._domainkey.example.com` TXT record.

Selectors are arbitrary DNS labels — `mail1`, `2026q1`, `hermes`, etc. — and there is no DKIM-defined upper bound on how many you publish concurrently.

Save flow

1. Validate form fields exist (when enabling DKIM)
 - Missing or out-of-set values -> `session.m = 20`, redirect, no DB write
2. `cfinclude dkim_set_settings.cfm`
 - a. UPDATE parameters child rows for the `smtpd_milters / non_smtpd_milters`:8891 entries (on or off)
 - b. UPDATE parameters2 rows for the nine OpenDKIM runtime directives
 - c. `cfinclude dkim_generate_config_file.cfm` – read `/opt/hermes/conf_files/opendkim.conf.HERMES`, REReplace the Canonicalization / On-* / SignatureAlgorithm placeholders, write `/etc/opendkim.conf`
 - d. `cfinclude dkim_generate_hosts.cfm` – regenerate `/opt/hermes/dkim/TrustedHosts` from `dkim_trusted_hosts`
 - e. `cfinclude dkim_generate_domains.cfm` – regenerate `/opt/hermes/dkim/ExemptDomains` from `dkim_bypass`
 - f. `cfinclude dkim_generate_keytable.cfm + dkim_generate_signingtable.cfm` – rebuild from `dkim_sign`
 - g. `cfinclude restart_opendkim.cfm` – docker exec inside `hermes_postfix_dkim` to restart BOTH opendkim instances
3. `cfinclude generate_postfix_configuration.cfm` – regenerate `main.cf` (`smtpd_milters` list reflects DKIM on/off) and reload Postfix
4. If DKIM was DISABLED: also flip off OpenDMARC milter rows, clear `FailureReports`, deactivate the DMARC report Ofelia job, regenerate `opendmarc.conf`, restart OpenDMARC
5. `session.m = 9` -> green "DKIM settings saved" alert on redirect

Add / Edit / Delete on the second card calls `dkim_generate_hosts.cfm` or `dkim_generate_domains.cfm` (whichever applies) plus `restart_opendkim.cfm` inline — Postfix is not reloaded since the milter chain itself did not change.

Files and containers touched

Path	Owner	Role
<code>config/hermes/var/www/html/admin/2/view_dkim_settings.cfm</code>	hermes_commandbox	The page
<code>config/hermes/var/www/html/admin/2/inc/get_dkim_settings.cfm</code>	hermes_commandbox	Loads current parameters / parameters2 / bypass / trusted-host values
<code>config/hermes/var/www/html/admin/2/inc/dkim_save_settings.cfm</code>	hermes_commandbox	Validates form, calls set + generate + restart chain; disables DMARC if DKIM off
<code>config/hermes/var/www/html/admin/2/inc/dkim_set_settings.cfm</code>	hermes_commandbox	UPDATES the parameters / parameters2 rows, regenerates all four config files, restarts OpenDKIM
<code>config/hermes/var/www/html/admin/2/inc/dkim_generate_config_file.cfm</code>	hermes_commandbox	Renders <code>/etc/opendkim.conf</code> from the template + DB
<code>config/hermes/var/www/html/admin/2/inc/dkim_generate_keytable.cfm</code>	hermes_commandbox	Rebuilds <code>/opt/hermes/dkim/KeyTable</code> from <code>dkim_sign</code>
<code>config/hermes/var/www/html/admin/2/inc/dkim_generate_signingtable.cfm</code>	hermes_commandbox	Rebuilds <code>/opt/hermes/dkim/SigningTable</code> from <code>dkim_sign</code>
<code>config/hermes/var/www/html/admin/2/inc/dkim_generate_hosts.cfm</code>	hermes_commandbox	Rebuilds <code>/opt/hermes/dkim/TrustedHosts</code> from <code>dkim_trusted_hosts</code>
<code>config/hermes/var/www/html/admin/2/inc/dkim_generate_domains.cfm</code>	hermes_commandbox	Rebuilds <code>/opt/hermes/dkim/ExemptDomains</code> from <code>dkim_bypass</code>
<code>config/hermes/opt/hermes/conf_files/opendkim.conf.HERMES</code>	hermes_commandbox (read) → hermes_postfix_dkim (live <code>/etc/opendkim.conf</code>)	Template with <code>HEADER-CANONICALIZATION</code> , <code>BODY-CANONICALIZATION</code> , <code>DEFAULT-ACTION</code> , etc. placeholders
<code>config/postfix-dkim/etc/opendkim-sign.conf</code>	hermes_postfix_dkim	Static config for the sign-only instance at <code>:8892</code> (no placeholders — relaxed/relaxed + rsa-sha256 are fixed for the re-injection signer)
<code>parameters</code> table (<code>inet:%:8891</code> rows under <code>smtpd_milters</code> and <code>non_smtpd_milters</code>)	hermes_db_server (hermes DB)	DKIM milter on/off
<code>parameters2</code> table (rows where <code>module='dkim'</code>)	hermes_db_server (hermes DB)	The nine OpenDKIM runtime settings
<code>dkim_sign</code> , <code>dkim_bypass</code> , <code>dkim_trusted_hosts</code> tables	hermes_db_server (hermes DB)	Per-domain keys, exempt-domain list, trusted-host list
<code>hermes_postfix_dkim</code> container	—	Runs both OpenDKIM instances and hosts the live config + key files

Path	Owner	Role
<code>hermes_unbound</code> container	—	Resolves every <code><selector>._domainkey.<domain></code> lookup

Failure semantics

Failure	Behavior
Missing form fields when enabling DKIM	<code>session.m = 20</code> , redirect, no DB write
Out-of-set value submitted for an Action / Canonicalization / Algorithm field	<code>session.m = 20</code> , redirect, no DB write
Empty entry on Add	<code>session.m = 13</code> , redirect, no DB write
Invalid syntax on Add / Edit	<code>session.m = 17</code> , redirect, no DB write
Duplicate entry on Add	<code>session.m = 14</code> , redirect, no DB write
<code>dkim_generate_config_file.cfm</code> write fails	Surfaces as <code>cfcatch</code> from the inline include — save aborts
<code>restart_opendkim.cfm</code> fails	Same path — Postfix is reloaded anyway in step 3, but DKIM service is left in the prior runtime state
<code>KeyTable</code> / <code>SigningTable</code> missing because no <code>dkim_sign</code> rows exist yet	OpenDKIM starts but signs nothing — outbound mail goes out unsigned

Related

- [SPF Settings](#) — the second authentication service whose result is consumed by DMARC; paired conceptually with DKIM as a "DNS-based outbound sender authentication" mechanism. SPF checks at envelope `MAIL FROM` time; DKIM checks header signatures after `DATA`. DKIM survives forwarding; SPF generally doesn't
- [DMARC Settings](#) — the policy layer that consumes DKIM (and SPF) results; disabling DKIM here automatically disables DMARC
- [ARC Settings](#) — the post-modification chain seal, which runs after the sign-only OpenDKIM at `:8892` so the ARC record covers the final outbound body
- [Trusted ARC Sealers \(M365\)](#) — for M365 customers whose downstream verifiers escalate when a Hermes-forwarded message's original DKIM signature breaks against the body-modified bytes
- [Perimeter Checks](#) — the SPF / DKIM / DMARC status card on Perimeter Checks links here for the per-service toggle
- [Domains \(Email Server\)](#) — where per-domain DKIM keys are generated, selectors chosen, and DNS TXT records exposed for publication

- [Domains \(Email Relay\)](#) — relay-mode domains can also sign outbound; same per-domain key UX
 - [Email Policies > Disclaimers](#) — documents the body milter that modifies outbound bodies before the sign-only OpenDKIM at `:8892` produces the final signature; the two-instance OpenDKIM design exists precisely because of this body modification
 - [DNS Resolver](#) — every `<selector>._domainkey.<domain>` lookup flows through `hermes_unbound`; resolver mode directly affects DKIM verification reliability
 - [System Certificates](#) — TLS on outbound delivery is independent of DKIM, but receivers that enforce strict transport security may surface DKIM failures more prominently in failure reports
-

Revision #48

Created 2026-05-31 12:52:22 UTC by Dino Edwards

Updated 2026-06-20 13:33:12 UTC by Dino Edwards