

Perimeter Checks

Perimeter Checks

Admin path: **Content Checks > Perimeter Checks** (`view_perimeter_checks.cfm`, `inc/get_perimeter_checks.cfm`, `inc/perimeter_save_settings.cfm`, `inc/generate_postfix_configuration.cfm`).

This page collects every **SMTP-time** check Hermes can apply before the message body is even read. Each control here writes a row (or toggles `enabled`) in the `parameters` table; on save, the `generate_postfix_configuration.cfm` include rebuilds `main.cf` from those rows via `postconf -e` and runs `postfix reload` inside `hermes_postfix_dkim`. There is no message-content inspection on this page — content scoring lives in [Anti-Spam Settings](#) and [Anti-Virus Settings](#), and runs only after the perimeter checks accept the connection.

Where perimeter checks sit in the flow

```
+-----+
| Remote SMTP peer |
+-----+-----+
      |
      v

+-----+-----+
| postscreen :25 (hermes_postfix_dkim) |
| - postscreen_access.cidr (whitelist/block) |
| - DNSBL scoring -> postscreen_dnsbl_sites |
| - pipelining / non-SMTP / bare-newline |
+-----+-----+
      | passes -> hand off
      v
+-----+-----+
```

```

| smtpd :25 |
| - smtpd_helo_required |
| - smtpd_client_restrictions |
| - smtpd_helo_restrictions |
| - smtpd_sender_restrictions |
| - smtpd_recipient_restrictions |
| (permit_mynetworks, permit_sasl_auth, |
| reject_unauth_destination, |
| reject_invalid_hostname, ..., |
| reject_rbl_client / DNSBL, |
| check_policy_service for SPF) |
| - message_size_limit |
+-----+-----+
| passes -> DATA accepted |
v
+-----+-----+
| Amavis / SpamAssassin / ClamAV (content) |
+-----+-----+

```

Perimeter Checks owns the postscreen knobs and the `smtpd*_restrictions` toggles. RBL list membership is split out to its own page — [RBL Configuration](#) — because the list is row-per-entry data, not a fixed set of switches.

The four cards on the page

1. Postscreen Settings

`postscreen` is Postfix's pre-queue connection filter — it sits in front of `smtpd` on port 25 and runs cheap protocol checks before any SMTP state machine is built. Three switches:

Switch	parameters row	Postfix directive	What it catches
Pipelining Detection	<code>postscreen_pipelining_enable</code>	<code>postscreen_pipelining_enable = yes/no</code>	Clients that send <code>EHLO</code> + <code>MAIL FROM</code> + <code>RCPT TO</code> in one TCP write before the server has finished its greeting — classic spambot shortcut

Switch	parameters row	Postfix directive	What it catches
Non-SMTP Command Detection	<code>postscreen_non_smtp_command_enable</code>	same	Clients that send something other than the SMTP verbs (typically HTTP <code>GET</code> from a misdirected scanner, or shellcode)
Bare Newline Detection	<code>postscreen_bare_newline_enable</code>	same	Clients that terminate lines with a bare <code>\n</code> instead of <code>\r\n</code> — RFC 5321 violation, very common in homebrew bot SMTP libraries

“ **Operational consequence.** Enabling any of these activates **greylisting-style deferral** for unknown clients. Mail from a well-behaved peer is delayed by one retry on first contact; mail from a peer that retries incorrectly (or not at all) is lost. The in-page callout warns about this explicitly. Leave these off until you have a reason to turn them on.

2. Message Limits

A single control: **Maximum Message Size (MB)**. The page displays the value in megabytes; on save it is multiplied by `1024*1024` and the integer byte count is written to the child row under the `message_size_limit` parent. Postfix enforces this at `DATA`-accept time and rejects with `552 5.3.4` if the message exceeds the limit.

Validation rejects zero, negative, and non-numeric input (`session.m = 3`).

3. SMTP Restrictions

The bulk of the page. The HELO toggle and seven recipient-side rejects each map to a child row under one of two parent parameters:

Toggle	Parent	Postfix directive	Rejects when...
Require HELO/EHLO	<code>smtpd_helo_required</code>	<code>smtpd_helo_required = yes</code>	Client tries to send <code>MAIL FROM</code> without first issuing <code>HELO</code> or <code>EHLO</code>
Reject Unauthorized Destination	<code>smtpd_recipient_restrictions</code>	<code>reject_unauth_destination</code>	Recipient domain is not a relay or hosted domain (open-relay protection — leave on)

Toggle	Parent	Postfix directive	Rejects when...
Reject Unauthorized Pipelining	<code>smtpd_recipient_restrictions</code>	<code>reject_unauth_pipelining</code>	Client pipelines commands without <code>EHLO</code> advertising support
Reject Invalid Hostname	<code>smtpd_recipient_restrictions</code>	<code>reject_invalid_hostname</code>	HELO/EHLO name is syntactically invalid (e.g. no dot)
Reject Non-FQDN Sender	<code>smtpd_recipient_restrictions</code>	<code>reject_non_fqdn_sender</code>	<code>MAIL FROM:</code> address has no fully-qualified domain
Reject Unknown Sender Domain	<code>smtpd_recipient_restrictions</code>	<code>reject_unknown_sender_domain</code>	Sender domain has neither MX nor A record in DNS
Reject Non-FQDN Recipient	<code>smtpd_recipient_restrictions</code>	<code>reject_non_fqdn_recipient</code>	<code>RCPT TO:</code> address has no fully-qualified domain
Reject Unknown Recipient Domain	<code>smtpd_recipient_restrictions</code>	<code>reject_unknown_recipient_domain</code>	Recipient domain has neither MX nor A record in DNS

The **DNSBL Threshold** field in the same card writes `postscreen_dnsbl_threshold` — the combined score that any single connecting IP must reach across all enabled DNSBL zones before postscreen rejects it. The shipped baseline is `3`. Per-zone weights are configured on [RBL Configuration](#); the threshold here is what those weights add up against. Validation requires an integer (`session.m = 2`).

“ **Order matters in Postfix.** The save routine does not let an admin reorder restrictions — the `order1` column in `parameters` is seeded at install time so that `permit_mynetworks` and `permit_sasl_authenticated` come first, then the `reject_unauth_destination` open-relay guard, then sender / recipient validation, then policy services. This is the canonical order; the UI only toggles which entries are active, not where they sit in the list.

4. Email Authentication (read-only status)

Three badges (SPF, DKIM, DMARC) showing whether each authentication service is wired into `smtpd_milters` / `smtpd_recipient_restrictions`, each with a small "Configure..." link to its dedicated page. This card is informational — toggling SPF/DKIM/DMARC on or off happens on:

- [SPF Settings](#) — child row under `smtpd_recipient_restrictions`
- [DKIM Settings](#) — milter at `inet:::8891` in `smtpd_milters`
- [DMARC Settings](#) — milter at `inet:::54321` in `smtpd_milters`

The DMARC row carries an additional note: DMARC requires SPF **and** DKIM to both be active. If either is disabled, the card surfaces "Requires both SPF and DKIM" inline.

Save flow

A single **Save & Apply Settings** click runs:

1. Validate `dnsbl_threshold` (integer) and `message_size_limit` (positive float)
 - Fail -> `session.m = 2` or `3`, cflocation back, no DB write
2. UPDATE parameters child rows for all toggles + values (applied = 2)
3. `cfinclude generate_postfix_configuration.cfm`
 - a. Copy `/opt/hermes/conf_files/main.cf.HERMES` -> `/etc/postfix/main.cf`
 - b. SELECT all enabled parents (child=2), join children (child=1)
 - c. Write `/opt/hermes/tmp/<trans>_postconf.sh` with one ``postconf -e "<directive> = <values>"`` line per parent
 - d. Append ``postfix reload``
 - e. `docker exec hermes_postfix_dkim /bin/bash <script>`
 - f. UPDATE parameters SET `applied=1, action='NONE'` WHERE `applied=2`
4. `session.m = 1` -> green "Settings Saved" alert on redirect
 - On failure -> `session.m = 4` with `cfcatch` detail surfaced in the alert

The reload is in-band — the page does not return until Postfix has reloaded (timeout: 240s).

The `parameters` dual-row pattern (perimeter-specific)

Every Postfix directive in Hermes is stored as **two-or-more linked rows** in the `parameters` table:

<code>child</code>	Role	What the <code>parameter</code> column holds
2	Parent (directive name)	The Postfix directive name (e.g. <code>smtpd_recipient_restrictions</code>)
1	Child (directive value)	One value the directive should emit (e.g. <code>reject_unauth_destination</code> , or <code>yes</code>)

Rows are linked by `parent_name` (child's `parent_name` matches parent's `parameter`) or by numeric `parent` (child's `parent` matches parent's `id`). The `order1` column sequences children inside a

parent so the generated `postconf -e` line emits values in a predictable order.

For perimeter checks, that means:

- `smtpd_helo_required` has **one** child row whose `parameter` is literally the string `yes` or `no` (toggle flips `enabled` on that one row).
- `smtpd_recipient_restrictions` has **many** child rows — one per restriction value. The toggle for each restriction flips `enabled` on its child row; the generator emits only `enabled=1` children.
- `message_size_limit` has one child row whose `parameter` is the literal byte-count string (e.g. `78643200`); the save handler rewrites that string on every save.

Failure semantics

Failure	Behavior
Invalid <code>dnsbl_threshold</code>	<code>session.m = 2</code> , redirect, no DB write
Invalid <code>message_size_limit</code>	<code>session.m = 3</code> , redirect, no DB write
<code>generate_postfix_configuration.cfm</code> throws	<code>session.m = 4</code> ; <code>session.postfix_error</code> is set to <code>cfcatch.message & cfcatch.detail</code> and surfaced under a small "Detail:" line in the red alert
<code>postfix reload</code> fails inside the container	Surfaces as a <code>cfcatch</code> from the <code>cfexecute</code> of the temp script — same <code>session.m = 4</code> path
<code>main.cf.HERMES</code> template missing in <code>/opt/hermes/conf_files/</code>	<code>cfcatch</code> on the template copy step — same path

The save is **not** transactional across the steps — if the SQL updates succeed but the reload fails, the DB state advances to `applied=2` and the next save attempt will pick those rows up and re-apply. The page does not strand partial state.

Files and containers touched

Path	Owner	Role
<code>config/hermes/var/www/html/admin/2/view_perimeter_checks.cfm</code>	<code>hermes_commandbox</code>	The page
<code>config/hermes/var/www/html/admin/2/inc/get_perimeter_checks.cfm</code>	<code>hermes_commandbox</code>	Loads parent IDs + current child values
<code>config/hermes/var/www/html/admin/2/inc/perimeter_save_settings.cfm</code>	<code>hermes_commandbox</code>	Validates form, updates <code>parameters</code> , calls the generator

Path	Owner	Role
<code>config/hermes/var/www/html/admin/2/inc/generate_postfix_configuration.cfm</code>	<code>hermes_commandbox</code>	Writes a temp <code>postconf -e</code> shell script, executes inside the postfix container, reloads Postfix
<code>config/hermes/opt/hermes/conf_files/main.cf.HERMES</code>	<code>hermes_commandbox</code> (read) → <code>hermes_postfix_dkim</code> (live <code>/etc/postfix/main.cf</code>)	Canonical template copied on every regen
<code>parameters</code> table	<code>hermes_db_server</code> (<code>hermes</code> DB)	Source of truth for every restriction and toggle
<code>hermes_postfix_dkim</code> container	—	Where <code>postconf -e</code> + <code>postfix reload</code> execute

Related

- [RBL Configuration](#) — the DNSBL list whose combined score is compared against the **DNSBL Threshold** on this page
- [Network Block/Allow](#) — the `postscreen_access` CIDR table consulted by postscreen before the DNSBL checks
- [Sender/Recipient Rules](#) — per-address override of perimeter-level rejects
- [SPF Settings](#), [DKIM Settings](#), [DMARC Settings](#) — the three authentication services whose status appears in card 4
- [Anti-Spam Settings](#) — content-time scoring that runs after a connection clears the perimeter
- [SMTP TLS Settings](#) — the cipher/protocol choices applied at the same `smtpd :25` listener
- [DNS Resolver](#) — every `reject_unknown_*_domain`, `reject_invalid_hostname`, and DNSBL query goes through `hermes_unbound`; resolver mode (recursive vs. forwarding) directly affects perimeter accuracy
- [Email flow](#) — full pipeline diagram

Revision #8

Created 2026-05-31 12:52:30 UTC by Dino Edwards

Updated 2026-05-31 14:01:23 UTC by Dino Edwards