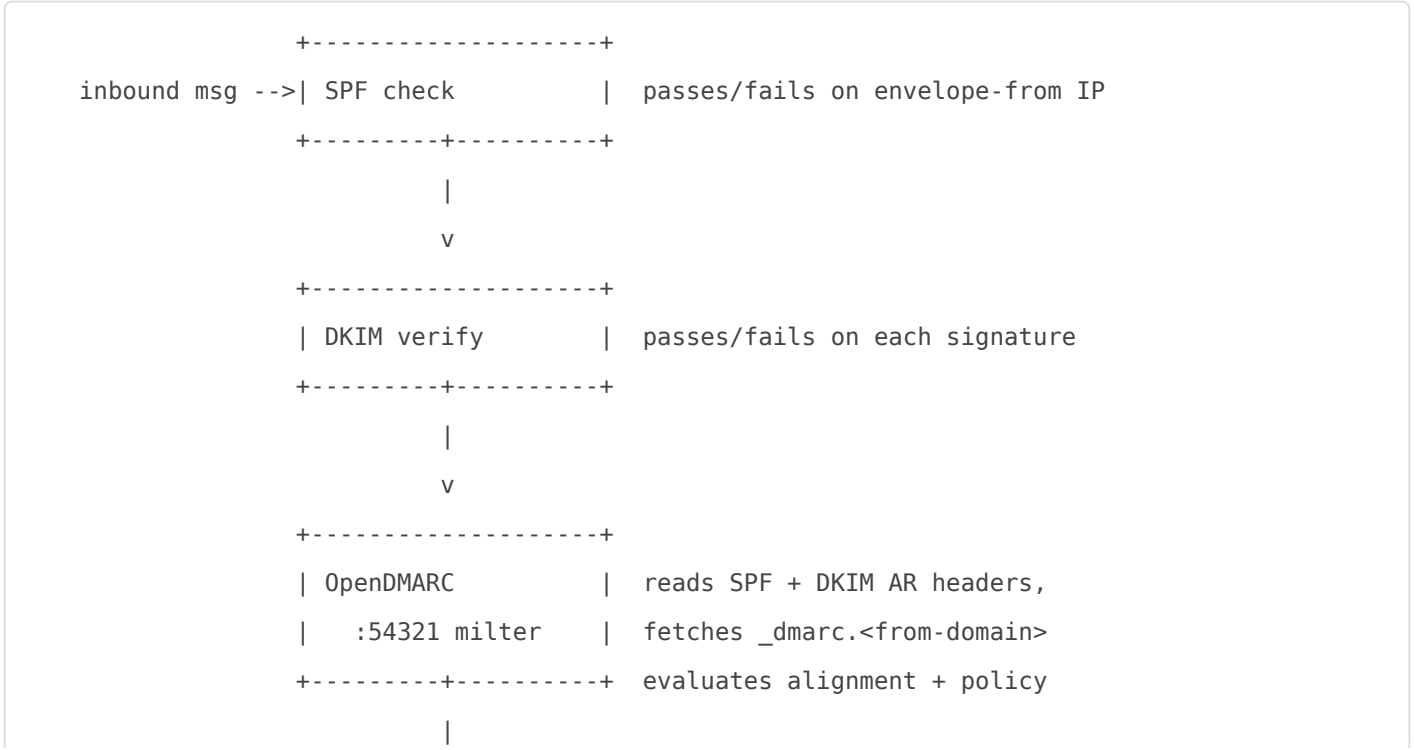
DMARC Settings

DMARC Settings

Admin path: **Content Checks > DMARC Settings** (`view_dmarc_settings.cfm`, `inc/get_dmarc_settings.cfm`, `inc/dmarc_save_settings.cfm`, `inc/dmarc_set_settings.cfm`, `inc/dmarc_generate_config_file.cfm`, `inc/dmarc_generate_reports_script.cfm`, `inc/restart_opendmarc.cfm`).

This page controls Hermes's OpenDMARC milter — both whether DMARC is evaluated on inbound mail and, when enabled, what happens to verdicts and whether daily aggregate reports are generated for the domains that publish a DMARC record. DMARC (RFC 7489) is the policy layer that sits on top of SPF and DKIM; a sender publishes a `_dmarc.<domain>` TXT record telling receivers what to do when neither SPF nor DKIM aligns with the From: header domain. Hermes is the receiver that does the work.

How DMARC fits the auth stack



```
                v
+--------------------+
| RejectFailures?    |
| -> reject / accept |
+--------------------+
```

A message **aligns** when its From: header domain matches the SPF-pass envelope-from domain OR the DKIM-pass `d=` domain. Relaxed alignment (the default) accepts org-domain match (`example.com` aligns with `mail.example.com`); strict alignment requires exact match. OpenDMARC reads the alignment results that SPF and DKIM have already written into the `Authentication-Results` header — both checks must therefore be active before DMARC is useful. The UI enforces this: enabling DMARC with SPF or DKIM disabled returns error 1.

Container and milter placement

Component	Detail
Container	`hermes_dmarc` (separate service, IPv4 `.111`)
Listen	`inet:54321@[0.0.0.0]` (`Socket` directive in `opendmarc.conf`)
Source	OpenDMARC daemon (Trusted Domain Project), packaged in the `hermes-dmarc` image
Milter chain	Postfix `smtpd_milters` AND `non_smtpd_milters` parents, child row `inet:<container>:54321` — toggle flips `enabled` on that row
DMARC report DB	`opendmarc` database on `hermes_db_server`, credentials in `system_settings` rows `mysql_username_opendmarc` / `mysql_password_opendmarc`
History file	`/etc/opendmarc/opendmarc.dat` inside `hermes_dmarc` (volume-mounted from `./config/opendmarc/etc/opendmarc/`)

The container exposes **no host ports** — Postfix reaches OpenDMARC internally at `inet:hermes_dmarc:54321`. The whitelist file path referenced by `DomainWhitelistFile` resolves to `/etc/opendmarc/whitelist.domains`, written by `inc/dmarc_generate_domains.cfm` from the `dmarc_domains` table on every save.

DMARC Settings card

Six controls drive `opendmarc.conf` directly via placeholder substitution into `/opt/hermes/conf_files/opendmarc.conf.HERMES`.

UI Control	`opendmarc.conf` directive	What it does
DMARC Enabled (YES/NO)	Milter chain toggle	Enables the `inet:%:54321` child row under `smtpd_milters` and `non_smtpd_milters`; OpenDMARC stops being consulted entirely when disabled
Reject Failures	`RejectFailures` (true/false)	When true, messages failing DMARC evaluation are rejected (or temp-failed if evaluation could not complete). When false, the message is accepted and only an `Authentication-Results` header records the verdict
Hold Quarantine Policy Messages	`HoldQuarantinedMessages` (true/false)	When true, messages from domains publishing `p=quarantine` that fail DMARC are routed to the Postfix `hold` queue for manual release/delete. When false (recommended), quarantine-policy messages are delivered with an `Authentication-Results` annotation and downstream scoring handles them
Generate Daily Failure Reports	`FailureReports` (true/false)	When true, OpenDMARC writes failure records to the history file and the daily Ofelia job converts them to RFC 6591 aggregate reports
Failure Reports From E-mail	`--report-email` flag on `opendmarc-reports`	RFC 6591 envelope From: for the outgoing report — must be a valid email address (validated by `IsValid("email", ...)`)
Failure Reports Reporting Organization	`--report-org` flag	Identifies your gateway as the report source — alphanumeric only (validation regex: `[^A-Za-z0-9]`)

OpenDMARC's `FailureReports` triggers reports only for domains that publish `p=quarantine` or `p=reject` (it never auto-reports for `p=none` unless `FailureReportsOnNone` is also set — Hermes does not expose that directive).

The "Reject Failures" UI hint and the OpenDMARC docs use the same language: messages that fail are **rejected** when policy is `reject`, **delivered with header** when policy is `none`, and **either held or flagged** when policy is `quarantine` (depending on `HoldQuarantinedMessages`).

> **Operational consequence — RejectFailures = true.** When this is on, OpenDMARC will respond `550 5.7.0` to messages from domains publishing `p=reject` that fail evaluation, and Postfix will refuse the message in-band. This catches forged messages but also catches legitimate forwarded mail from

> senders whose original SPF / DKIM chain breaks at an upstream forwarder. If you start seeing legitimate forward-from-mailing-list mail bounce, the fix is to add the originating domain to the Whitelisted Domains card below — not to disable Reject Failures globally.

Whitelisted Domains card

Rows from the `dmarc_domains` table (`id`, `domain`, `note`, `type`) write to `/etc/opendmarc/whitelist.domains`. OpenDMARC reads that file via `DomainWhitelistFile` and bypasses DMARC evaluation entirely for any matching From: domain — no alignment check, no policy enforcement, no failure report. Use for trusted senders with known broken DMARC, partner domains that forward through aggregators that strip headers, or legacy mailing lists.

Only domain names are accepted; IP addresses are rejected by the add handler. Domains are validated by the same regex used elsewhere in Hermes (e.g. error 17: "The entry is not a valid domain"). Bulk add is supported one-per-line in the textarea.

DMARC report generation (daily aggregate / RUA)

When **Generate Daily Failure Reports** is enabled, `dmarc_set_settings.cfm` calls `dmarc_generate_reports_script.cfm` which renders `/opt/hermes/scripts/dmarc_report_script.sh` with credentials and identifiers substituted into placeholders (`DATABASE-SERVER`, `DATABASE-USER`, `DATABASE-PASSWORD`, `REPORTING-EMAIL`, `REPORTING-ORGANIZATION`, `POSTMASTER-EMAIL`) and writes the result to `/opt/hermes/schedule/dmarc_report_script.sh` (chmod +x).

An Ofelia job named `hermes-dmarc-report` runs the script daily at 02:30:

```
[job-exec "hermes-dmarc-report"]
schedule:  0 30 02 * * *
container: hermes_dmarc
command:   /opt/hermes/schedule/dmarc_report_script.sh
```

The script does three things in sequence:

1. **`opendmarc-import`** — drains `/etc/opendmarc/opendmarc.dat` (the per-message verdict log OpenDMARC writes) into the `opendmarc` MariaDB database

2. `opendmarc-reports` — generates RFC 6591 aggregate XML reports for the prior 24h interval and emails one report per sender domain to the `rua=` address that domain published in DNS
3. `opendmarc-expire` — drops records older than the retention window from the database

The script also emits a Net::SMTP success/failure notification to the `postmaster` address (from `system_settings`). The Perl one-liner passes the postmaster address through an environment variable rather than direct string interpolation — Perl's default array sigil `@` treats `@deeztek.net` as an array dereference and silently loses the domain part. Passing via `$ENV{POSTMASTER_ARG}` avoids the trap (the fix landed as issue #215). The notification is also skipped entirely when `postmaster` is not a valid email address (e.g. bare local-part like `postmaster`) — this prevents queue pollution with undeliverable bounces.

SMTP delivery uses `hermes_postfix_dkim:10026` (the post-amavis re-injection port) — using `:25` would re-process the report through the inbound pipeline and could re-trigger DMARC evaluation on the report itself.

When **Generate Daily Failure Reports** is disabled (or DMARC itself is disabled), the save handler:

- Deletes `/opt/hermes/schedule/dmarc_report_script.sh`
- Sets `ofelia_jobs.active = '2'` on the `hermes-dmarc-report` job and regenerates `/etc/ofelia/config.ini` via `ofelia_generate_config.cfm`

Forensic (RUF) reports

Forensic (per-failure) reports are intentionally **not** generated by Hermes. They are privacy-noisy (they include redacted copies of failing messages), receivers rarely publish a `ruf=` address, and the modern operational consensus is that aggregate (RUA) reports give operators the visibility they need without the per-message exhaust. The `FailureReportsBcc` / `FailureReportsSentBy` / `CopyFailuresTo` directives in `opendmarc.conf.HERMES` are left commented and not exposed in the UI.

ARC interaction

Hermes also runs an ARC sealer (`hermes_openarc`) on the same authentication stack. When Hermes modifies a message body (External Sender Banner, disclaimer injection, signature injection, S/MIME or PGP rewrap), the original sender's DKIM body hash no longer matches the current body — DMARC alignment is lost on the modified copy. ARC preserves the pre-modification verdict in a sealed chain so downstream receivers configured to trust Hermes can still rescue DMARC alignment. See ARC Settings and the Trusted ARC Sealers — M365 guide for the receiver-side

configuration. Hermes is the authoritative auth boundary for every domain it serves; customer downstream MX allowlisting is the standard remedy when ARC trust is not in play.

Save flow

```
1. View page submits action=save_settings or add_domain / edit_domain / delete_domain
2. dmarc_save_settings.cfm validates:
     - SPF + DKIM both enabled (error 1 if not)
     - rejectfailures / holdquarantinedmessages / failurereports are true|false (error 20)
     - if failurereports=true: report_email present + valid (errors 2, 3)
                               report_org present + alphanumeric (errors 4, 5)
3. dmarc_set_settings.cfm UPDATEs:
     - parameters.enabled on the inet:%:54321 child row (smtpd + non_smtpd)
     - parameters2.value2 on FailureReports / RejectFailures / HoldQuarantinedMessages
       (module = 'dmarc')
     - parameters2.value2 on report_email / report_org (when reports enabled)
4. dmarc_generate_config_file.cfm:
     - Copies opendmarc.conf.HERMES to /opt/hermes/tmp/<trans>_opendmarc.conf
     - Substitutes FAILURE-REPORTS, REJECT-FAILURES, HOLD-QUARANTINE-MESSAGES placeholders
     - Backs up /etc/opendmarc/opendmarc.conf -> opendmarc.HERMES
     - Moves the rendered file into place
5. dmarc_generate_reports_script.cfm (if reports enabled):
     - Renders dmarc_report_script.sh, chmod +x
     - Enables ofelia_jobs row for hermes-dmarc-report, regenerates Ofelia config
   (else: deletes the script, disables the Ofelia row)
6. restart_opendmarc.cfm: docker container restart hermes_dmarc
7. generate_postfix_configuration.cfm: postconf -e the milter list, postfix reload
8. session.m = 9 -> green "DMARC settings saved successfully. Postfix reloaded." alert
```

Failure semantics

Failure	Behavior
SPF or DKIM not enabled when DMARC=YES	session.m = 1, redirect, no DB write
`report_email` empty	session.m = 2
`report_email` invalid	session.m = 3

Failure	Behavior
`report_org` empty	session.m = 4
`report_org` contains non-alphanumeric	session.m = 5
Missing required form fields	session.m = 20
Delete Domains clicked with nothing selected	session.m = 11
Add Domain with empty Domain field	session.m = 13
Add Domain with invalid format	session.m = 17
Add Domain with duplicate	session.m = 14 (single) or `_exists` alert (bulk)

Files and containers touched

Path	Owner	Role
`config/hermes/var/www/html/admin/2/view_dmarc_settings.cfm`	`hermes_commandbox`	The page
`config/hermes/var/www/html/admin/2/inc/dmarc_*.cfm`	`hermes_commandbox`	Validate / save / generate / restart
`config/hermes/opt/hermes/conf_files/opendmarc.conf.HERMES`	`hermes_commandbox` (read) -> `hermes_dmarc` (live `/etc/opendmarc/opendmarc.conf`)	Canonical template
`config/hermes/opt/hermes/scripts/dmarc_report_script.sh`	`hermes_commandbox` (read) -> rendered into `/opt/hermes/schedule/` (executed in `hermes_dmarc`)	Daily aggregate report script
`/etc/opendmarc/whitelist.domains`	`hermes_dmarc`	Generated from `dmarc_domains` table on every save
`/etc/opendmarc/opendmarc.dat`	`hermes_dmarc`	Per-message verdict history; drained nightly by `opendmarc-import`
`opendmarc` MariaDB DB	`hermes_db_server`	Holds imported verdicts that `opendmarc-reports` reads
`parameters` / `parameters2` tables (`module='dmarc'`)	`hermes_db_server` (`hermes` DB)	Source of truth for every directive
`system_settings` rows `mysql_username_opendmarc` / `mysql_password_opendmarc`	`hermes_db_server`	DB creds for the report script (managed via `update_opendmarc_db_creds.cfm`)
`ofelia_jobs` row `hermes-dmarc-report`	`hermes_db_server`	Daily report scheduler entry

Related

- Perimeter Checks — the SMTP-time card whose Email Authentication badge shows DMARC's wired-up status and the "Requires both SPF and DKIM" callout
- SPF Settings — the alignment input for the envelope From: side
- DKIM Settings — the alignment input for the signature `d=` side
- ARC Settings — preserves the DMARC verdict across body-modifying forwarding hops
- Trusted ARC Sealers — M365 — receiver-side configuration to trust Hermes's ARC seal
- Anti-Spam Settings — runs after DMARC and can promote a DMARC-fail message to higher spam score
- Score Overrides — per-rule weight changes
- DNS Resolver — every `_dmarc` TXT lookup goes through `hermes_unbound`; resolver mode (recursive vs. forwarding) directly affects DMARC accuracy and report timing
- Email flow — full pipeline diagram with milter placement

Revision #12
Created 2026-05-31 12:52:23 UTC by Dino Edwards
Updated 2026-06-11 15:04:31 UTC by Dino Edwards