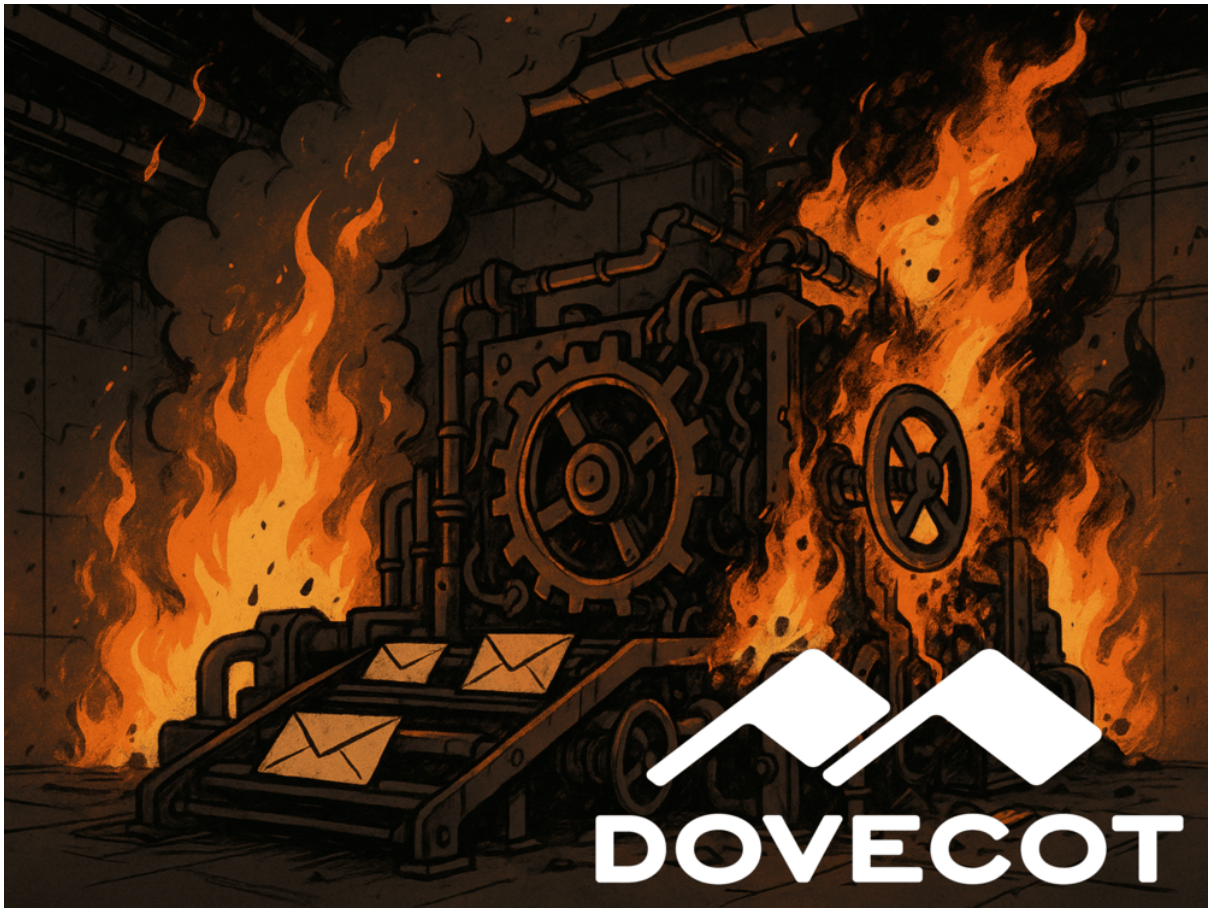
Brekende Wijzigingen

Dovecot 2.3 naar 2.4 upgraden in Debian Stable

Willem L. Middelkoop
4 juni 2025



Vorige week liep ik tegen onverwachte problemen aan tijdens een routinematige onderhoudsprocedure op de e-mailinfrastructuur van mijn bedrijf. We verwerken dagelijks duizenden e-mails en gebruiken Dovecot om onze klanten toegang te geven tot hun berichten. Dat kwam allemaal tot stilstand toen ik updatete naar versie 2.4, die belangrijke wijzigingen bevat... o jee!

TL;DR: Dovecot 2.3 → 2.4 upgrade op Debian

- **Dovecot 2.4 introduceert ingrijpende wijzigingen**: Configuratiebestanden van 2.3 zijn incompatibel. De software zal niet starten tenzij je ze herschrijft.
- **Nieuwe config-syntax en variabelensysteem**: Parameters zoals ‘ssl_cert‘, ‘mail_location‘ en ‘%d‘ hebben een andere opmaak en betekenis gekregen.

- **Geen geautomatiseerd migratiepad**: Handmatige controle en herschrijving van de configuratie is vereist—vooral bij gebruik van SQL-backends, Sieve of aangepaste authenticatie.
- **Aanbeveling**: Lees de officiële upgrade handleiding grondig door voordat je upgradet, en test configs afzonderlijk.

Fluitje van een cent

In afwachting van enkele beveiligingsupdates en om enkele compatibiliteitsproblemen op te lossen, nam ik de beslissing om enkele pakketten bij te werken op een Debian GNU/Linux-server die verantwoordelijk is voor de e-mailservices van mijn bedrijf. Deze specifieke distributie van GNU/Linux staat bekend om zijn stabiliteit en betrouwbaarheid. Er was geen reden om problemen te verwachten en ik startte de upgrade in de verwachting dat deze binnen enkele minuten zou zijn voltooid. Gewoon "een fluitje van een cent", wat kan er misgaan..?

'dovecot_config_version' vereist

Het grootste deel van de server kwam zoals verwacht bijna onmiddellijk weer online, waardoor de services zo weinig werden verstoord dat zelfs het uptime monitor alarm geen enkele storing registreerde - behalve voor Dovecot. Dit specifieke type software is verantwoordelijk voor IMAP- en POP3-toegang tot mailboxen. Het past als een fijn afgestemd tandwiel in een grotere setup waar andere programma's zoals Postfix erop vertrouwen dat het werkt om inkomende e-mail te accepteren. Het feit dat Dovecot niet startte, veroorzaakte een reeks problemen.

Het eerste wat je als serverbeheerder doet, is de serverdaemons controleren ('systemctl status dovecot') gevolgd door het inspecteren van de logs, bijvoorbeeld door 'journalctl -u dovecot' uit te voeren. Dat onthulde al snel dat Dovecot weigerde te starten omdat een config-parameter ontbrak: 'dovecot_config_version'.

Voeg de parameter gewoon toe aan de config in '/etc/dovecot/dovecot.conf' zou je denken? Nou... NEE. Het blijkt dat de makers van Dovecot de fout met een goede reden hebben toegevoegd: de nieuwe 2.4-versie van Dovecot is incompatibel met eerdere 2.3-configuratiebestanden; wat betekent dat je veel verschillende parameters moet herschrijven. Het zijn niet alleen de namen van de config-variabelen, het introduceert ook een geheel nieuwe syntax voor het uitbreiden van instellingsvariabelen. Een oude variabele waarde zoals '%d' wordt nu '%{user | domain}'. De lijst met wijzigingen is lang en er is geen geautomatiseerde manier om je config te migreren. Stel je mijn gezicht voor toen ik me de enorme omvang van mijn probleem realiseerde.

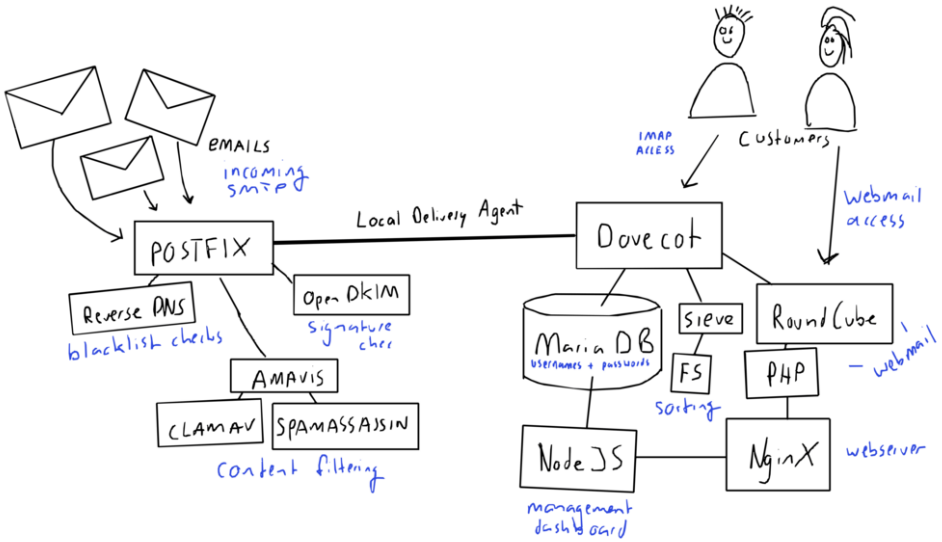
Geïntegreerde complexiteit: tandwielen in een machine

Het punt met e-mailservices is dat het *moeilijk* is. Niet omdat één component moeilijk is, maar omdat je voor het uitvoeren van een complete e-mailstack veel verschillende onderdelen nodig hebt die naadloos samenwerken. Hier zijn enkele van de onderdelen waaruit de e-mailservice van mijn bedrijf bestaat:

- **Postfix:** De primaire SMTP mail transfer agent (MTA), deze communiceert met andere mailservers om e-mail te ontvangen en te verzenden. Het moet weten welke

e-mailadressen geldig zijn voor het ontvangen van e-mails en wie gemachtigd is om nieuwe berichten te verzenden.

- **Amavis-filter:** Alle e-mails die door de server worden verwerkt, gaan door het amavis-filtermechanisme dat individuele berichten controleert met behulp van SpamAssissin en ClamAV.
- **SpamAssasin:** Met behulp van een complexe set regels en Bayesiaanse analyses worden berichten automatisch gescand op de waarschijnlijkheid dat het spam is.
- **ClamAV:** Berichten worden gescand op virussen. Bijlagen worden uitgepakt en bestand voor bestand geanalyseerd.
- **Reverse DNS-blacklists:** Naast het scannen van de berichten, voert de mailserver ook een achtergrondcontrole uit op externe servers waarmee hij communiceert. E-mailheaders worden geparseerd en individuele hops worden gecontroleerd op hun reputatie.
- **OpenDKIM:** Sommige berichten bevatten een 'domeinsleutel'-handtekening, waaruit blijkt dat hun afzender gemachtigd is. Berichten met headers worden gecontroleerd, terwijl anderen worden ondertekend met behulp van een filteringsdienst die is gekoppeld aan Postfix.
- **Sieve:** Inkomende berichten worden automatisch gesorteerd met behulp van gebruikersregels, bijvoorbeeld door bepaalde berichten in specifieke mappen te laten bezorgen. Het moet weten welke regels van toepassing zijn op welke adressen.
- **MariaDB / MySQL:** De e-mailadressen, geldige logins, gebruikersnamen, wachtwoordhashes en aliassen worden opgeslagen in een gecentraliseerde database. Specifieke SQL-query's zijn geconfigureerd om dingen te controleren zoals ”is dit een geldig wachtwoord?”.
- **Nginx:** De e-mailserver heeft ook een webserver die het beheer van accounts mogelijk maakt en toegang biedt tot e-mail via webmail.
- **NodeJS:** De beheerconsole maakt gebruik van een NodeJS-backend om serverbeheerders en clients in staat te stellen hun adressen en wachtwoorden in te stellen.
- **Roundcube**: De server gebruikt een veelgebruikt webmailprogramma om clients toegang te geven tot hun e-mail via de browser.
- **PHP:** De webmail vereist PHP om te draaien, inclusief enkele afhankelijkheden om het te laten communiceren met Dovecot (met behulp van IMAP) en MariaDB (met behulp van SQL).
- **Dovecot:** Dan is er Dovecot, de eigenlijke software die eindgebruikers toegang geeft tot hun e-mail. Het stelt geauthenticeerde gebruikers in staat om toegang te krijgen tot de e-mails die zijn opgeslagen op de schijven van de infrastructuur met behulp van protocollen zoals IMAP of POP3.



Een complete mailserverstack bestaat uit vele kleinere componenten die nauwkeurig op elkaar zijn afgestemd

Deze onderdelen zijn allemaal fijn op elkaar afgestemd, zoals tandwielen in een complexe machine passen ze precies. Hun configuratie is afgestemd op hun specifieke rol, waarbij ze vaak een interface hebben met meerdere andere onderdelen. Je kunt je voorstellen dat een onverwachte verandering in een van deze tandwielen (zoals Dovecot) behoorlijk wat verstoring kan veroorzaken!

Er was geen snelle oplossing beschikbaar, geen geautomatiseerde configuratie-upgrade. Aangezien deze update relatief nieuw is, hadden de meeste grote taalmodellen (AI) ook geen antwoord. ChatGPT genereerde simpelweg suggesties op basis van de oude 2.3-documentatie. Om het probleem op te lossen, moest ik *'ouderwetse Linux-servervaardigheden'* toepassen: handmatig een nieuwe, op maat gemaakte configuratie voor Dovecot 2.4 maken.

Stop om gegevensverlies te voorkomen

Zodra ik de aard van het probleem begreep, stopte ik onmiddellijk de meeste e-mailservices (bijv. door de motor volledig te stoppen). Dit voorkomt gegevensverlies aangezien e-mails niet meer worden verwerkt. Dit zorgt ervoor dat andere mailservers e-mails in de wachtrij plaatsen, soms met 4xx SMTP-fouten (zoals 421, 451, 454 of 455). Het ontwerp van het wereldwijde e-mailprotocol is geschikt voor tijdelijke storingen en andere servers zullen het later gewoon opnieuw proberen.

De config herschrijven door RTFM

Het foutbericht in de logs verwees me naar de officiële Dovecot documentatie, en daar was het: een gelig gemarkeerde WAARSCHUWING die de configuratiewijzigingen beschrijft. Het is een lang document: als PDF telt het 18 pagina's. Op een normale dag zou het enige moeite kosten om dit door te nemen - stel je voor dat je dit moet doen terwijl het spreekwoordelijke huis in brand staat!

De configuratie van Dovecot bestaat uit verschillende bestanden die allemaal delen van het gedrag van de software instellen. Zoals hoe authenticatie wordt uitgevoerd, of hoe het integreert met een Sieve-filter, of waar het de gebruikersnaam- en wachtwoordhashes voor login vandaan haalt. De wijzigingen die zijn geïntroduceerd in Dovecot 2.4 hebben bijna invloed op alle parameters, aangezien de makers een nieuwe syntax voor servervariabelen hebben geïntroduceerd.

Syntax voor variabelennotatie

Voorheen gebruikte de configuratie variabelennotatie zoals '%u' als tijdelijke aanduiding voor een gebruikersnaam. De nieuwe syntax staat toe dat variabelen worden doorgestuurd naar een modifier, waardoor normalisatie en dingen zoals subtekenreeksselectie mogelijk zijn. Bijvoorbeeld: om een e-maildomein van een gebruikersnaam te krijgen, gebruik je niet langer '%d' maar gebruik je de 'user'-variabele als basis om deze door te sturen naar de domeinmodifier met behulp van deze syntax: '%{user | domain}'.

Toen ik deze nieuwe taal eenmaal begreep, waardeerde ik het kunstwerk dat het met zich meebrengt echt. Het is een verbetering ten opzichte van de oudere afgekorte variabelennamen, aangezien je niet langer een woordenboek nodig hebt om dingen op te zoeken: in plaats van iets vaags zoals '%r' lees je nu '%{remove_ip}'. Het is inherent leesbaarder, iets dat ik echt waardeer.

Encryptie: 'ssl_cert' wordt 'ssl_server_cert_file'

Andere wijzigingen zijn onder meer de manier waarop de config verwijst naar externe bestanden die worden gebruikt voor encryptie. Voorheen had je een instelling 'ssl_cert' die het eigenlijke SSL/TLS-certificaat als tekenreeks kon bevatten. In versie 2.4 heb je de variabele 'ssl_server_cert_file' die in plaats daarvan een pad naar het certificaat op het bestandssysteem aanneemt. **Het is niet alleen hernoemen, de aard van de waarde is ook nieuw!** Om de configuratie te repareren, moet je elk van deze bijgewerkte parameters doorlopen en hun impact op je specifieke setup bepalen. Het is tijdrovend en foutgevoelig.

De 'mail_location'-instelling splitsen

Nieuw in 2.4 is hoe de 'mail_location' is opgesplitst in meerdere kleinere variabelen. Voorheen definieerde één variabele waar e-mail op schijf werd opgeslagen. Je moet dit nu opsplitsen in twee variabelen 'mail_driver' en 'mail_path', **EN** de nieuwe servervariabelensyntax gebruiken. Als je een van deze fout hebt, zal alle e-mail verdwijnen (geloof me!). Afhankelijk van je specifieke opslagconfiguratie wil je het gebruikersnaamdeel dubbel controleren (bijv. neem je het volledige e-mailadres op of alleen het voorvoegsel zonder domeinnaam?).

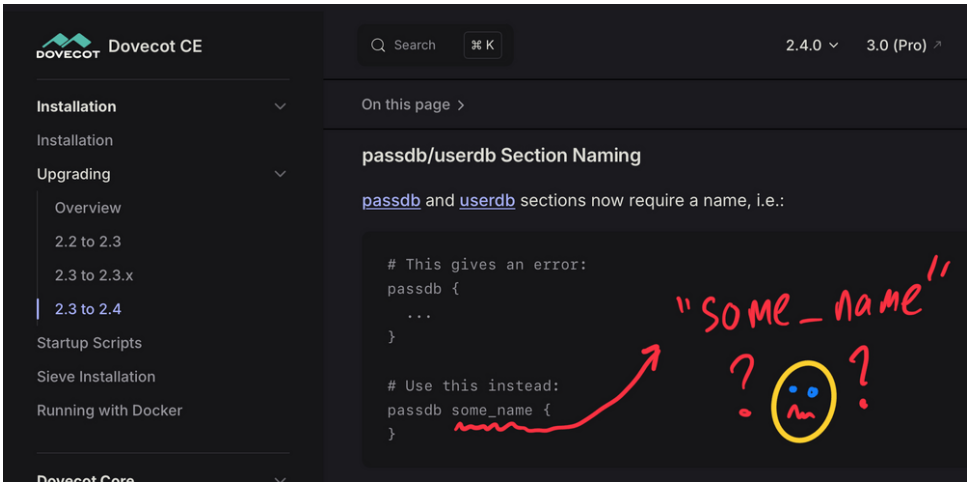
```
# Dovecot 2.3 config:
# mail_location = maildir:/var/vmail/%d/%u

# Corresponds to 2.4 syntax:
mail_driver = maildir
mail_path = /var/vmail/%{user | domain }/%{user | username}
```

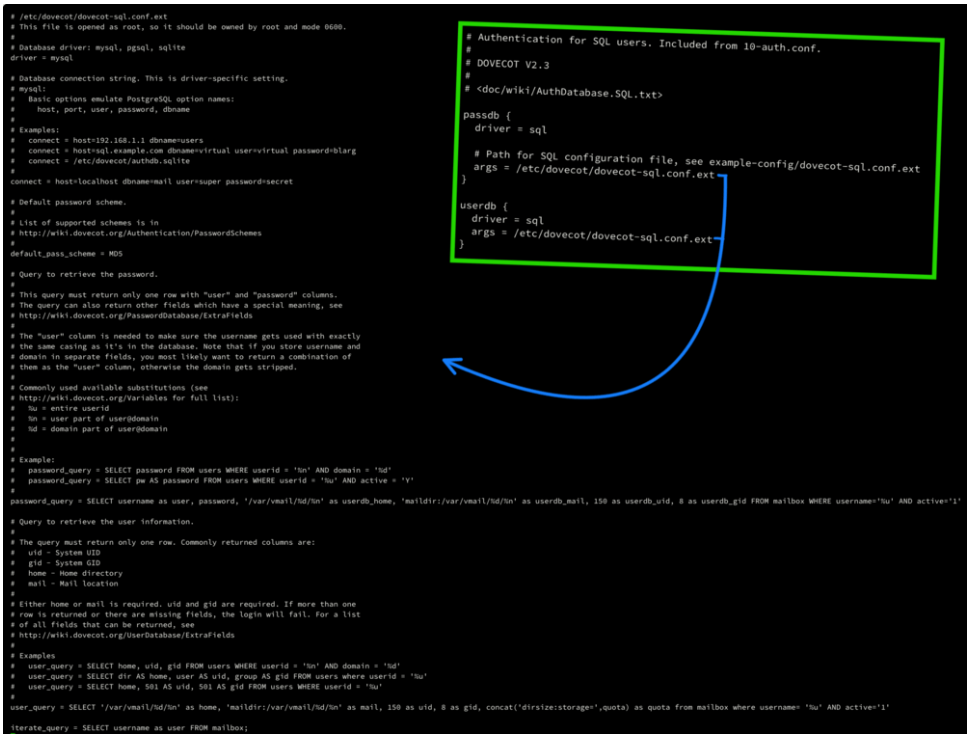
Migratie van mail_location naar mail_driver en mail_path met behulp van de nieuwe servervariabelesyntaxis

Dovecot met SQL-backend: passdb en userdb

Als je Dovecot-installatie werkt met een externe database (of opslag) om gebruikersnamen en wachtwoorden te authenticeren, moet je de 'auth-sql.conf.ext' in '/etc/dovecot/conf.d/' herschrijven, die wordt aangeroepen/opgenomen door het bestand 10-auth.conf. In Dovecot 2.4 is er een nieuwe manier om passdb- en userdb-instellingen te segmenteren. In de Dovecot 2.3-setup gebruikte ik een extern bestand om de databasequery's en verbindingsreeks op te geven. Tijdens de authenticatie worden de meerdere query's gebruikt om 1) gebruikers te authenticeren en 2) gebruikersspecifieke instellingen zoals het pad van de e-mailopslag te laden.



De upgrade-instructies voor userdb en passdb zijn op het eerste gezicht zeer nuttig /s



Dovecot 2.3 SQL-backend setup (auth-sql.conf.ext en dovecot-sql.conf.ext)

In de nieuwe 2.4-versie moet je de parameters herstructureren zodat ze worden opgenomen in secties die een backend-type-indicator bevatten direct achter de trefwoorden 'userdb' en 'passdb' in de config. Dus, als je een MySQL-database als backend gebruikt, moet je 'userdb' veranderen in 'userdb sql'. Je kunt de parameters van de verbindingsreeks direct in het bestand 'auth-sql.conf.ext' opnemen, zodat alle relevante parameters in één configuratiebestand staan. Om de complexiteit te verminderen, heb ik het userdb-gedeelte herschreven om de 'statische' database driver te gebruiken: het genereren van configuratiewariabelen zoals het e-mailpad van de gebruiker op basis van de gebruikersnaam van de gebruiker. Zie de speciale documenten.

```
# Authentication for SQL users. Included from auth.conf.
#
# DOVECOT 2.4
#
# <https://doc.dovecot.org/latest/core/config/auth/databases/sql.html>

# Database driver: mysql, pgsql, sqlite
sql_driver = mysql

# Database connection string. This is driver-specific setting.
mysql localhost {
    user = super
    password = secret
    dbname=mail
}

passdb sql {
    default_password_scheme = MD5
    query = SELECT username as user, password FROM mailbox WHERE username='%{user}' AND active='1'
}

# If you don't have any user-specific settings, you can avoid the user_query
# by using userdb static instead of userdb sql, for example:
# <https://doc.dovecot.org/latest/core/config/auth/databases/static.html>
userdb static {
    fields {
        uid = 150
        gid = 8
        home = /var/vmail/%{user | domain}/%{user | username}
    }
}
```

Dovecot 2.4 SQL-backend setup (auth-sql.conf.ext) - nu veel compacter dan zijn voorganger. Er is nu slechts één SQL-query omdat de userdb-sectie nu de statische backend gebruikt

Sieve-plug-in

Als je setup Sieve gebruikt om inkomende e-mails te sorteren, moet je je setup aanpassen met betrekking tot de locatie van het script, de standaardinstellingen en het gedrag. De nieuwe Dovecot 2.4-versie introduceert een concept genaamd ”Scriptopslag” waarmee dynamisch geladen scripts kunnen worden uitgevoerd voor verschillende gebruikers bij verschillende gebeurtenissen. Onze setup omvat een eenvoudig standaard sieve-script dat e-mail die als spam is gemarkeerd automatisch sorteert naar een map met de naam Spam. De nieuwe syntax maakt een compactere config mogelijk:

```
##
## Settings for the Sieve interpreter
##
## DOVECOT 2.3
##

# Do not forget to enable the Sieve plugin in 15-lda.conf and 20-lmtp.conf
# by adding it to the respective mail_plugins= settings.

plugin {
  # The path to the user's main active script. If ManageSieve is used, this the
  # location of the symbolic link controlled by ManageSieve.
  sieve = ~/.dovecot.sieve

  # The default Sieve script when the user has none. This is a path to a global
  # sieve script file, which gets executed ONLY if user's private Sieve script
  # doesn't exist. Be sure to pre-compile this script manually using the sievec
  # command line tool.
  # --> See sieve_before fore executing scripts before the user's personal
  #     script.
  sieve_default = /var/lib/dovecot/sieve/default.sieve

  # Directory for :personal include scripts for the include extension. This
  # is also where the ManageSieve service stores the user's scripts.
  sieve_dir = ~/sieve
}
```

Sieve's configuratie in Dovecot 2.3 (90-sieve.conf) met nu verouderde variabelen sieve_default en sieve_dir

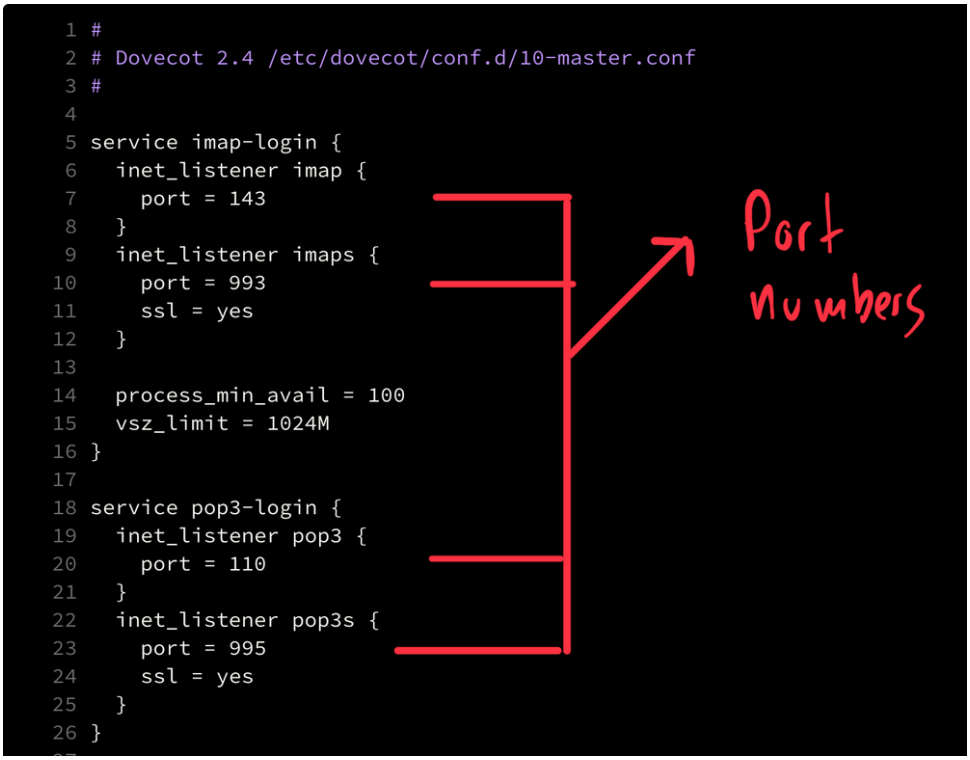
```
1  # /etc/dovecot/conf.d/90-sieve.conf
2  #
3  # DOVECOT 2.4
4  #
5  # Sieve default settings loading global script
6
7  sieve_script default {
8      type = default
9      name = default
10     driver = file
11     path = /var/lib/dovecot/sieve/default.sieve
12 }
```

Sieve's configuratie in Dovecot 2.4 (90-sieve.conf)

De nieuwe config testen

Je zou in de verleiding kunnen komen om de server gewoon opnieuw te starten nadat je de 2.4-configuratie opnieuw hebt gedefinieerd. Maar je moet heel voorzichtig zijn, want een verkeerde configuratie kan ertoe leiden dat clients de toegang tot alle e-mails verliezen en/of dat ze gedwongen worden hun volledige mailboxen opnieuw te downloaden. In plaats daarvan koos ik voor een geleidelijke aanpak waarbij ik de IMAP- (en POP3-) poortnummers veranderde in iets willekeurigs. Op deze manier kon ik de nieuwe setup helemaal zelf testen terwijl ik de daadwerkelijke klanten ervan weerhield verbinding te maken met de server. Dit stelde me in staat om enkele typefouten te corrigeren, een deel van de nieuwe servervariabelensyntax te verfijnen en enkele vreemde logberichten

te verwerken. Je kunt de poortnummers van de server regelen vanuit het bestand '10-master.conf'.



Stel de IMAP- en POP-nummers van de server in om privé/gefaseerde debugging van de nieuwe configuratie mogelijk te maken

Conclusie

Storingen zoals deze zijn zeldzaam, dankzij de stabiliteit van Debian GNU/Linux, maar niet onmogelijk. Dagelijkse back-ups en een getest noodherstelplan zijn aanwezig voor de Lemmid Online e-mailservice van mijn bedrijf. Hoewel een volledige failover deze keer niet nodig was, bleken eerdere oefeningen cruciaal te zijn om de configuratie snel aan te passen.

Dit incident was een herinnering dat, zelfs met automatisering en AI, diepgaande systeemkennis, voorbereiding en handmatig debuggen essentiële hulpmiddelen blijven wanneer kritieke infrastructuurcomponenten onverwachts en zonder geautomatiseerde migratiepaden veranderen.

Voorbeeldconfiguratie Dovecot 2.4

Als je dit bericht vindt door te zoeken naar een oplossing voor je eigen kapotte mailserver, vind je deze Dovecot 2.4-voorbeeldconfiguratiebestanden misschien nuttig. Ik vond ze tijdens mijn eigen reparatiewerkzaamheden en ze kunnen je wat tijd besparen bij het uitzoeken van de nieuwe syntax: https://source.willem.com/dovecot-2.4-sample-config/