max planck institut informatik

MAX-PLANCK-GESELLSCHAFT

Max Planck Institute for Software Systems

Exim MTA in der Praxis

Patrick Cernko

pcernko@mpi-sb.mpg.de

Max-Planck-Institute
für Informatik & Softwaresysteme
Saarbrücken

2. Juli 2007

Übersicht

- 1. Teil: Theorie
  - Allgemeines, Entstehung, Hintergründe
  - Einführung in die Konfiguration
    Strukturierung des Config-Files
  - Konzepte von Exim
    - Expansions-Ausdrücke: *Exim's Programmiersprache*
    - ACLs & Router im Detail: *„Wer darf? Wo geht's hin?"*
    - Kommandozeile & Tools: *Aufruf, Tests, Debugging*
- 2. Teil: Praxis
  - Praxis: Szenario MPI
    Komplexes und doch übersichtliches Mail-System

Allgemeines, Entstehung, Hintergründe

- 1995 entwickelt von Philip Hazel
  University of Cambridge in England, PCRE Library
- Ursprünglich „**EX**perimental **I**nternet **M**ailer”
- Single-Binary Design Modell
  vgl. Sendmail™
- Aktuelle Version: 4.67, OpenSource, GPL
- Distributionen: Debian, Ubuntu, Red Hat, SUSE, Gentoo, FreeBSD, Solaris (CSW)
  Sourcecode portiert auf viele weitere Unix-Derivate

Einführung in die Konfiguration

- **Eine** Konfigurationsdatei
  - *alles weitere: Includes oder Tabellen&Datenbanken*
  - *Lang & gut lesbar*
  - *BerkleyDBs für internen Status (Retry, Callout, ...)*

Einführung in die Konfiguration

- **Eine** Konfigurationsdatei
  - *alles weitere: Includes oder Tabellen&Datenbanken*
  - *Lang & gut lesbar*
  - *BerkleyDBs für internen Status (Retry, Callout, ...)*

  - Globale Einstellungen
    - Macro-Definitionen
    - Datenstrukturen
      Listen für Local-Domains, Relay-Domains, Relay-Hosts, . . .
    - ACL-Zuordnungen
    - Queueing, Logging, Ports, SSL/TLS

Einführung in die Konfiguration

- Eine Konfigurationsdatei
  - ACL-Definitionen
    - *Entscheidung über Erfolg eines (SMTP-)Kommandos*
    - *Beispiele:*

`RCPT␣TO`	Open Relay?
`DATA`	Mail-Syntax-Check

Einführung in die Konfiguration

- Eine Konfigurationsdatei
  - ACL-Definitionen
    - *Entscheidung über Erfolg eines (SMTP-)Kommandos*
    - *Beispiele:*

      `RCPT␣TO` Open Relay?

      `DATA` Mail-Syntax-Check
  - Router-Definition
    - *Wohin damit?*
    - *Entscheidung anhand priorisierter Liste von Möglichkeiten*

Einführung in die Konfiguration

- Eine Konfigurationsdatei
  - Transports
    - *Wie kommt die Mail zum Ziel?*
    - *Exekutive zur Legislative Router*

Einführung in die Konfiguration

- Eine Konfigurationsdatei
  - Transports
    - *Wie kommt die Mail zum Ziel?*
    - *Exekutive zur Legislative Router*
  - Retry-Konfiguration
    - *Wie oft & wie viele Zustellungsversuche?*
    - *Beispiel:*

```
# Domain            Error    (Senders)   Retries
*@+local_domains      *                  F,2h,5m;    F,8h,15m; F,8h,1h;   F,4d,4h
*                     *       senders=:  F,1h,30m
*                     *                  F,2h,15m;      G,16h,1h,1.5;     F,4d,6h
```

Einführung in die Konfiguration

- Eine Konfigurationsdatei
  - Rewrite-Konfiguration
    - *Umschreiben/„Korrigieren" von E-Mail-Adressen*
    - `to`, `from`, `To`, `From`, `CC`, `BCC`, `Reply-To`, `Sender`
    - *Sehr flexibel dank Exim-Funktionen*
    - *Hazel:* ***Do not use!***

Einführung in die Konfiguration

- Eine Konfigurationsdatei
  - Rewrite-Konfiguration
    - *Umschreiben/„Korrigieren" von E-Mail-Adressen*
    - `to`, `from`, `To`, `From`, `CC`, `BCC`, `Reply-To`, `Sender`
    - *Sehr flexibel dank Exim-Funktionen*
    - *Hazel:* ***Do not use!***
  - Authenticators
    - *Client- & Server-Authentifizierung*
    - *Backends: Files, (SQL-)DBs, SASL (⇒ passwd, NIS, LDAP,* ***PAM****)*
    - *Bei uns: Remote-IMAP & Radius dank PAM*

Exim: Expansions-Ausdrücke

- „Programmiersprache” von Exim
  „. . . the expanded value . . . ”

Exim: Expansions-Ausdrücke

- „Programmiersprache” von Exim
  „. . . the expanded value . . . ”
- Großes Set an Funktionen & Variablen
  - *PCRE Library*

    ```
    ${sg{$h_x-spam-mpi-notes-tag:}{\N[\t\n]\N}{ }}
    ```
  - *String-Manipulation*

    ```
    .../${substr{6}{2}{$tod_logfile}}/${substr{11}{2}{$tod_log}}
    ```

    20070*02* 2007-07-02 *10*:45:00
  - Lookup *in Files, DNS, DBs, NIS, LDAP, . . .*

    ```
    data = ${lookup mysql {SELECT email FROM login \
                           WHERE user="${quote_mysql:$local_part}"}{$value}fail}
    ```
  - *Programme, Sockets, Embedded Perl*

Exim: Expansions-Ausdrücke

- „Programmiersprache” von Exim
  „. . . the expanded value . . . ”
- Großes Set an Funktionen & Variablen
  - *PCRE Library*

    ```
    ${sg{$h_x-spam-mpi-notes-tag:}{\N[\t\n]\N}{ }}
    ```
  - *String-Manipulation*

    ```
    .../${substr{6}{2}{$tod_logfile}}/${substr{11}{2}{$tod_log}}
    ```

    20070*02* 2007-07-02 *10*:45:00
  - Lookup *in Files, DNS, DBs, NIS, LDAP, . . .*

    ```
    data = ${lookup mysql {SELECT email FROM login \
                           WHERE user="${quote_mysql:$local_part}"}{$value}fail}
    ```
  - *Programme, Sockets, Embedded Perl*
- **Statische** Einstellungen werden **dynamisch**

Exim: Expansions-Ausdrücke (Beispiele)

```
tls_certificate = \
  ${lookup {$interface_address}                \
           lsearch{CONFDIR/certificate_map} \
           {$value}                            \
           fail}

certificate_map:
139.19.1.25 /etc/ssl/mail.mpi-inf.mpg.de.pem
139.19.1.24 /etc/ssl/mail.mpi-sws.mpg.de.pem
```

- *Auf den ersten Blick nur eine Datei spezifizierbar, dank Expansions-Ausdruck jedoch dynamisch*
- *Statt* `fail` *auch Default oder Fallback möglich*

Exim: Expansions-Ausdrücke (Beispiele)

```
tls_certificate = \
  ${lookup dnsdb                                \
           {ptr=$interface_address}             \
           {/etc/ssl/$value.pem}                \
           fail}
```

- *DNS-Service ersetzt Datei.*
- *Bei mehreren DNS-Ergebnissen evtl. String-Substitution & reguläre Ausdrücke notwendig*

Exim: ACLs

- „Access Control Lists” für (SMTP-)Befehle
  Alle SMTP-Befehle (`RCPT␣TO`, `DATA`, `HELO`, . . .)
  plus „connect” & non-smtp
  Beispiel:

```
acl_smtp_rcpt = acl_check_rcpt
begin acl
  acl_check_rcpt:
    accept
      hosts = +relay_from_hosts

    deny
```

Exim: ACLs

- „Access Control Lists” für (SMTP-)Befehle
  Alle SMTP-Befehle (`RCPT␣TO`, `DATA`, `HELO`, ...)
  plus „connect” & non-smtp
  Beispiel:

```
acl_smtp_rcpt = acl_check_rcpt
begin acl
  acl_check_rcpt:
    accept
      hosts = +relay_from_hosts

    deny
```

- Vielzahl von Primitiven & Informationsquellen
  - `recipients, senders, domain, authenticated, hosts, verify = sender(/callout), ...`
  - *Sehr flexibel durch*
    `condition =` *`Expansions-Ausdruck`*

Exim: ACLs (Beispiele)

```
acl_check_mail:
  deny # deny servers, which do not start with a proper HELO/EHLO
    message = no HELO given before MAIL command
    condition = ${if def:sender_helo_name {no}{yes}}
  ...
  warn
    message = Forged IP detected in HELO: $sender_helo_name
    log_message = Forged IP detected in HELO: $sender_helo_name
    condition = ${if eq{$sender_helo_name}{$interface_address} {yes}{no}}
  ...

acl_check_rcpt:
  ...
  accept # accept relaying after authentication
    authenticated = *
  ...
  accept
    domains = +relay_to_domains
    endpass
    verify = recipient
  ...
```

Exim: ACLs

- Nicht nur Ablehnung!
  Logging, Header-Ergänzung, „State-Control”, temp. Ablehnung, Blackhole

Exim: ACLs

- Nicht nur Ablehnung!
  Logging, Header-Ergänzung, „State-Control”, temp. Ablehnung, Blackhole
- Spam- & Virenfilter
  - *Einbindbar über fertige Primitive oder vorbereitete Sockets*

  $\Rightarrow$ *Ablehnung zur SMTP-Zeit*

  oder: *Nur Tagging möglich (*`warn`*-Statement)*

Exim: ACLs

- Nicht nur Ablehnung!
  Logging, Header-Ergänzung, „State-Control”, temp. Ablehnung, Blackhole
- Spam- & Virenfilter
  - *Einbindbar über fertige Primitive oder vorbereitete Sockets*

  ⇒ *Ablehnung zur SMTP-Zeit*

  oder: *Nur Tagging möglich (`warn`-Statement)*
- Ideal für eigene Anpassungen und Anforderungen
  - *Logging von Mails mit best. Eigenschaften*
  - *Privilegien-Finetuning*

  z.B. veraltete/unerwünschte MUAs ablehnen.

Exim: Router

- Entscheidungskette für das Zustellungsziel und Transportmedium
  - *„Order matters!”*
  - *Indirekt verwendet in ACLs:* `verify = recipient`

Exim: Router

- Entscheidungskette für das Zustellungsziel und Transportmedium
  - *„Order matters!”*
  - *Indirekt verwendet in ACLs:* `verify = recipient`
- Vielzahl von Primitiven & Informationsquellen (wie ACLs), jedoch anderer Zugriffsfokus
  - *API ausgerichtet auf Routing*

Exim: Router (Beispiel)

```
dnslookup:
  debug_print = "R: dnslookup for $local_part@$domain"
  driver = dnslookup
  domains = ! +local_domains
  transport = remote_smtp
  same_domain_copy_routing = yes
  # ignore private rfc1918 and APIPA addresses
  ignore_target_hosts = 0.0.0.0 : 127.0.0.0/8 :\
                        192.168.0.0/16 : 172.16.0.0/12 :\
                        10.0.0.0/8 : 169.254.0.0/16 :\
                        255.255.255.255
  no_more
```

Exim: Router

- Definiert Transport zur Zustellung
  - Lokale Zustellung
    `mbox, maildir, mbx, mailstore`
  - Remote Zustellung
    `smtp, autoreply`
  - Programm-Pipes
    spamc, procmail, maildrop, cyrus, mailman

Exim: Router

- Definiert Transport zur Zustellung
  - Lokale Zustellung
    `mbox, maildir, mbx, mailstore`
  - Remote Zustellung
    `smtp, autoreply`
  - Programm-Pipes
    spamc, procmail, maildrop, cyrus, mailman
- Router ohne Transport: `redirect`
  - *z.B. Aliases, Forward/Filter*

  ⇒ *Ergebnis des Routings ist wieder ein Fall für's Routing.*

  oder: *Spezielle Parameter für (Filter-)Ergebnisse*

Exim: Router (Beispiel 2)

```
userforward:
  debug_print = "R: userforward for $local_part@$domain"
  driver = redirect
  domains = +local_domains
  check_local_user
  file = $home/.forward
  require_files = $local_part:$home/.forward
  no_verify
  no_expn
  check_ancestor
  allow_filter
  forbid_smtp_code = true
  directory_transport = address_directory
  file_transport = address_file
  pipe_transport = address_pipe
  reply_transport = address_reply
  skip_syntax_errors
  syntax_errors_to = real-$local_part@$domain
  syntax_errors_text = \
    This is an automatically generated message...
```

Exim: Kommandozeile & Tools

- Reichhaltige API
  Single-Binary, Daemon-Startup, Debugging, Management (Mail-Queue)

Exim: Kommandozeile & Tools

- Reichhaltige API
  Single-Binary, Daemon-Startup, Debugging, Management (Mail-Queue)
- Viele Testmöglichkeiten
  - *Expansions-Ausdrücke*
  - *Filter (→* `userforward`*)*
  - *Relaying*
  - *Config-Values*
  - *Adressverifikation*
  - *Rewriting*

Exim: Kommandozeile & Tools (Beispiel)

```
$ exim4 -brw pcernko@europa.mpi-sb.mpg.de
  sender: pcernko@mpi-sb.mpg.de
    from: pcernko@mpi-sb.mpg.de
      to: pcernko@mpi-sb.mpg.de
      cc: pcernko@mpi-sb.mpg.de
     bcc: pcernko@mpi-sb.mpg.de
reply-to: pcernko@mpi-sb.mpg.de
env-from: pcernko@mpi-sb.mpg.de
  env-to: pcernko@mpi-sb.mpg.de
```

```
$ exim4 -brw pcernko@imap.mpi-sb.mpg.de
  sender: pcernko@mpi-sb.mpg.de
    from: pcernko@mpi-sb.mpg.de
      to: pcernko@mpi-sb.mpg.de
      cc: pcernko@mpi-sb.mpg.de
     bcc: pcernko@mpi-sb.mpg.de
reply-to: pcernko@mpi-sb.mpg.de
env-from: pcernko@mpi-sb.mpg.de
  env-to: pcernko@imap.mpi-sb.mpg.de
```

Exim: Kommandozeile & Tools

- *Sendmail*™ kompatibel
  - `/usr/lib/sendmail -> ../sbin/exim4`
  - *Die meisten Optionen werden „ignoriert”. Nur zur Kompatibilität gegenüber alten Programmen*

`-B<type>` This is a Sendmail option for selecting 7 or 8 bit processing. Exim is 8-bit clean; it ignores this option.

Exim: Kommandozeile & Tools

- *Sendmail*™ kompatibel
  - `/usr/lib/sendmail -> ../sbin/exim4`
  - *Die meisten Optionen werden „ignoriert”.*
    Nur zur Kompatibilität gegenüber alten Programmen

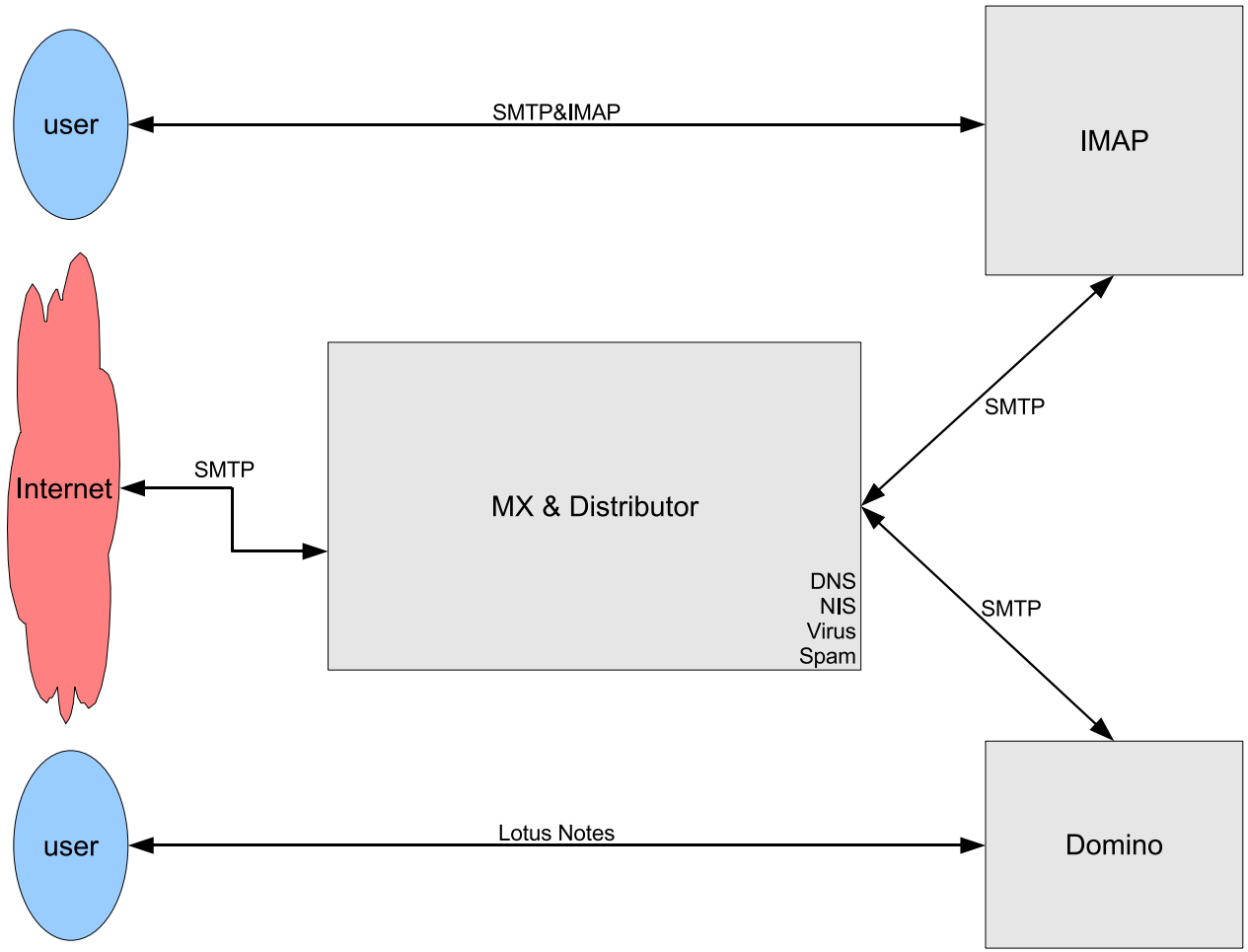
`-B<type>` This is a Sendmail option for selecting 7 or 8 bit processing. Exim is 8-bit clean; it ignores this option.

- weitere Tools
  - `exim_dumpdb, exim_fixdb, exim_tidydb` zum Modifizieren der internen BerkleyDBs
    z.B. Retry-Timeout nach AMaViS-Neustart löschen
  - `exim_checkaccess`
    Wrapper für Relay-Tests

Übersicht (2. Teil)

- 1. Teil: Theorie
- 2. Teil: Praxis
- Praxis: Szenario MPI
  Mailserver-Infrastruktur für zwei Forschungsinstitute
  - Blick über das System
    Vom Monolith zu den Funktionseinheiten
  - Funktionseinheiten
    Details zu den Komponenten des Systems
  - Spezielle Features mit Exim
    Außergewöhnliche Anforderungen einfach realisiert

Szenario MPI: Überblick

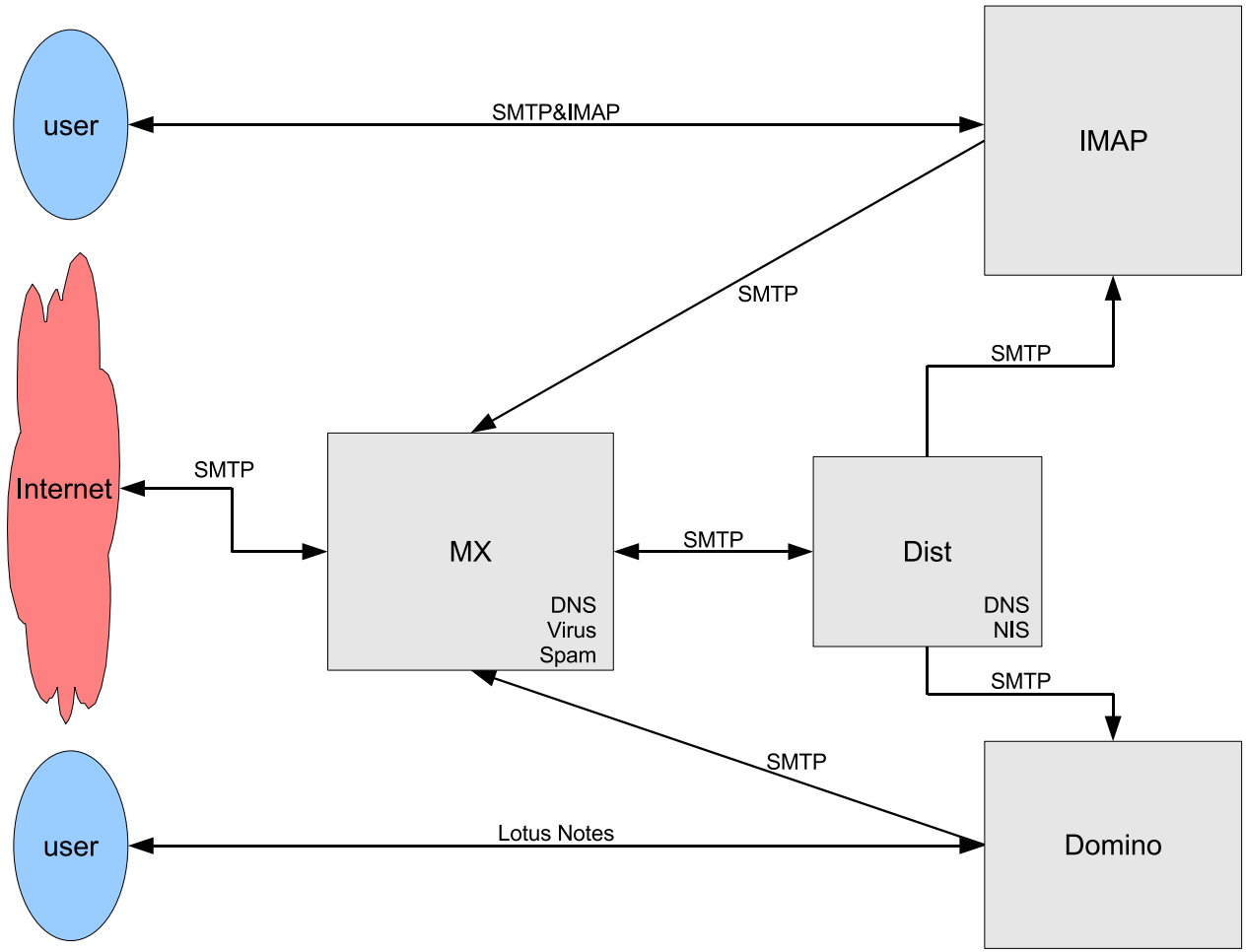


Vogelperspektive

Zwei Backends zur Speicherung

Zentrales Mail-Exchanger-System

Szenario MPI: Überblick

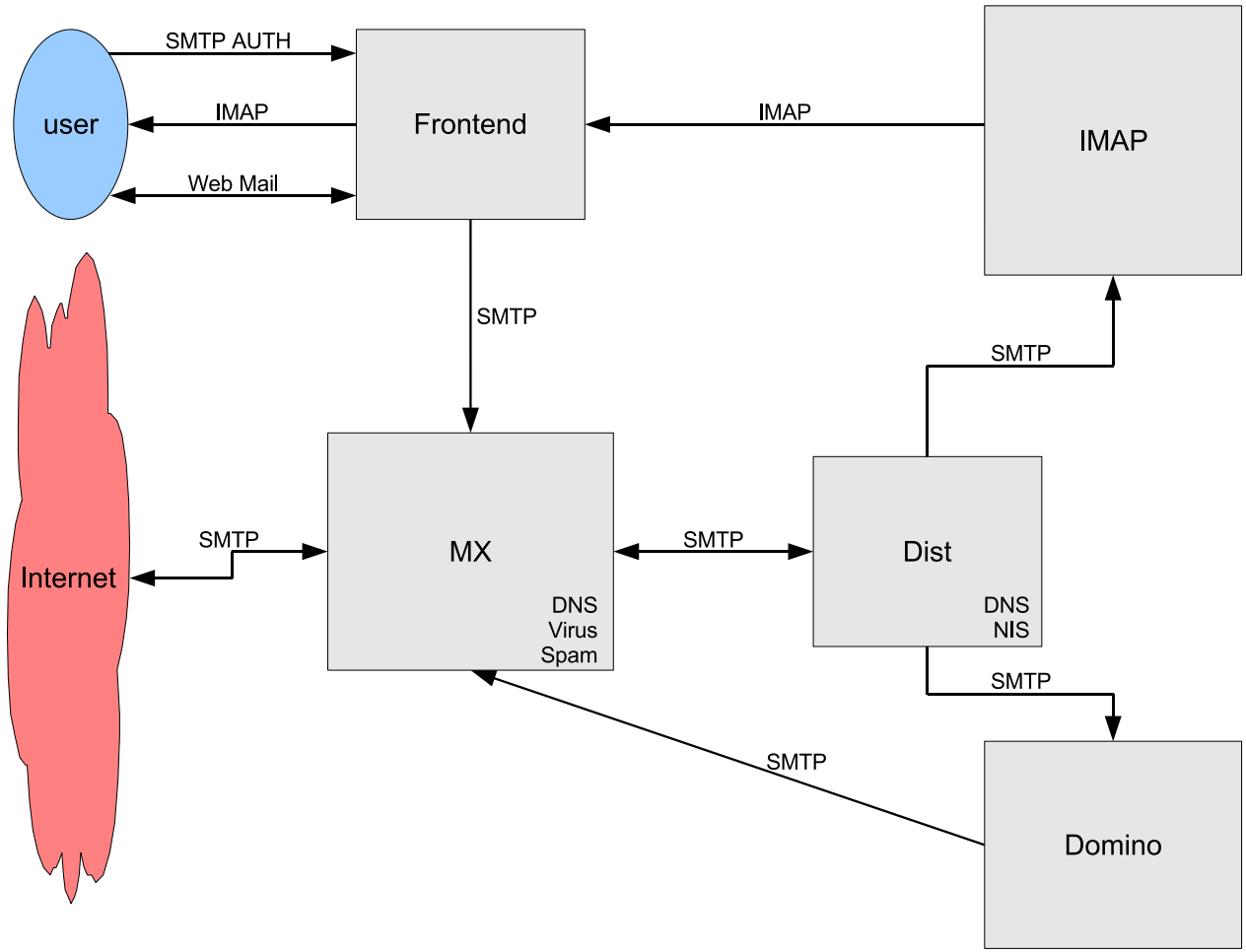


Service-Separation

„Desinfektion” zuerst

Keine Benutzerdaten (NIS) in der DMZ

Szenario MPI: Überblick

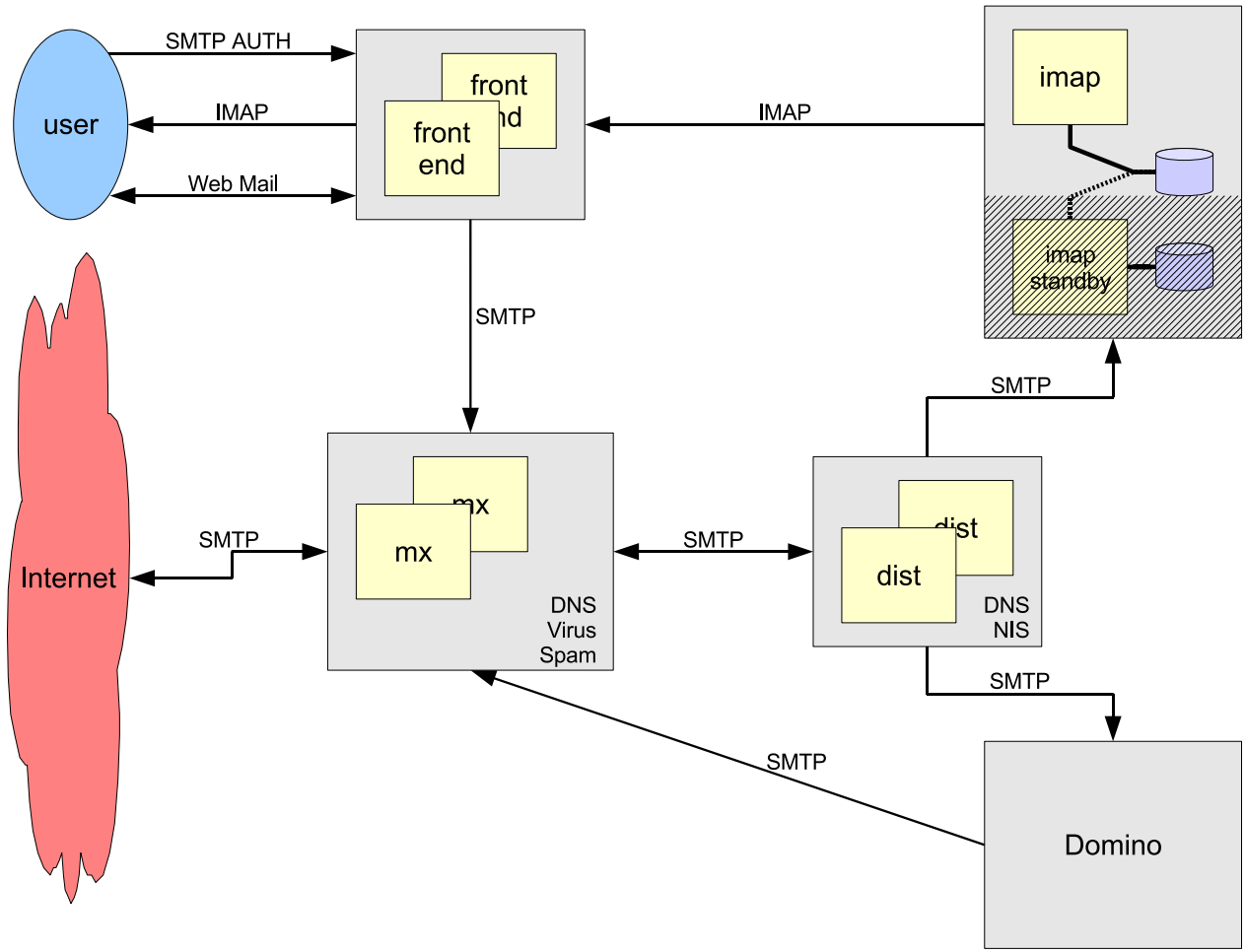


Transparenz

Entkopplung Mail-Storage ↔ User-Interaktion

Prinzip DMZ weitergeführt

Szenario MPI: Überblick

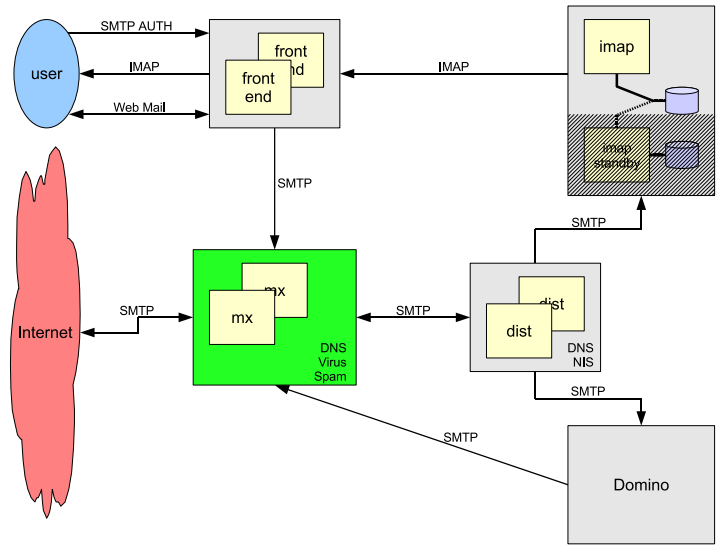


Redundanz & Ausfall-Absicherung

Vollständig Redundante Systeme, falls möglich

Hot-Standby als Alternative

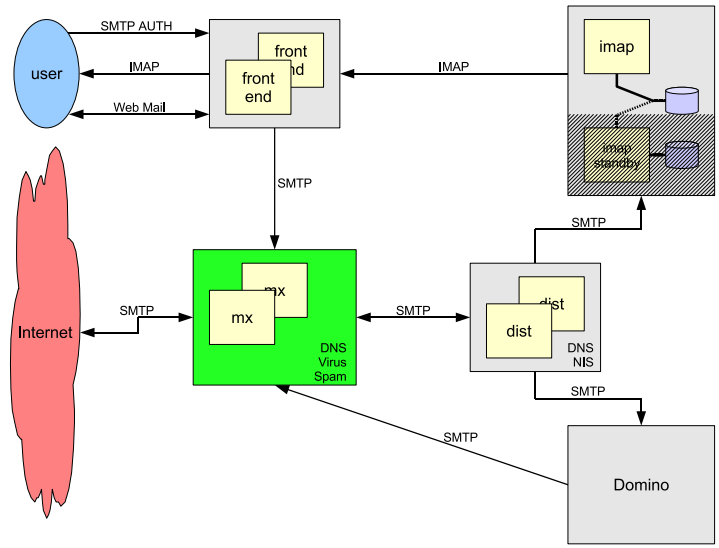
MPI: Funktionseinheit Mail-Exchanger



- Schnittstelle mit Außenwelt
  Jede Mail geht hier raus oder rein.
- Virus-Desinfektion
  - *Ankommende Viren aufhalten*
    → Empfänger informieren
  - *Ausgehende Viren aufhalten*
    → Absender informieren

⇒ Keine Virus-Benachrichtigung nach außen (= Unbekannte)

MPI: Funktionseinheit Mail-Exchanger

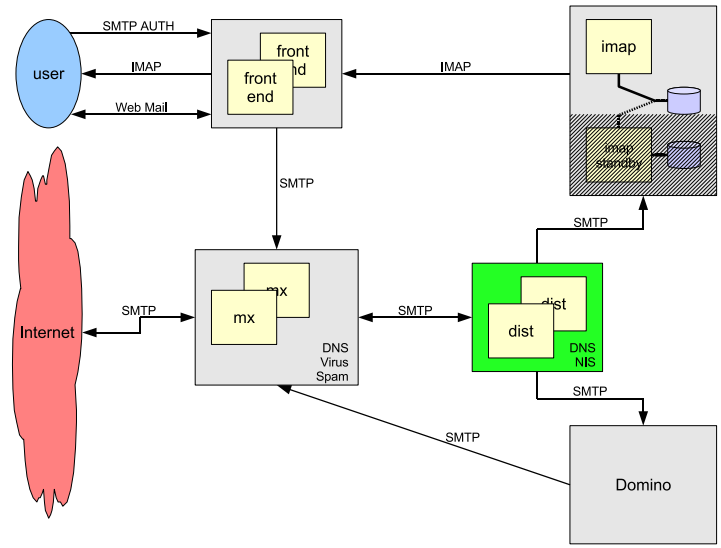


- Spambewertung
  - *Private E-Mail gestattet*

  $\Rightarrow$ *Keine Zensur möglich („Telekommunikationsgesetz”)*

  $\Rightarrow$ *Sorgfältiges Tagging für Benutzer*
- Wartungspuffer
  - *Trotz Wartungsarbeiten erreichbar*
  - *Keine Abhängigkeit von Retry-Regeln anderer*
  - *Bonus: Schnellere Alias- & Verteiler-Expansion*

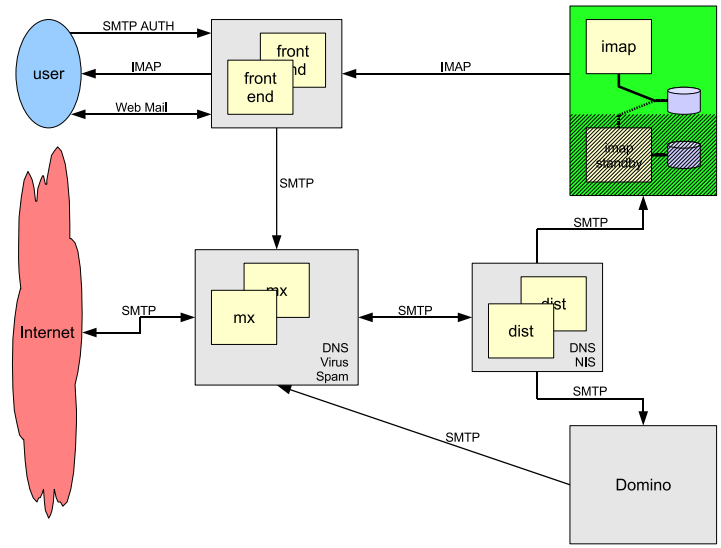
MPI: Funktionseinheit Mail-Dists



- User-orientiertes Routing
  - ♦ *Aliases & Verteiler*
  - ♦ *Zuständiges Endsystem*
  - ♦ *Forwarding*
- Standalone dank selbständiger Config-Aktualisierung
  - ♦ *Web-Interface & separater Fileserver*

  ⇒ *Automatische Pullmechanismen*
- Wartungspuffer

MPI: Funktionseinheit IMAP-Server



- Ablösung Unix-Mail System
  `/var/spool/mail + procmail`

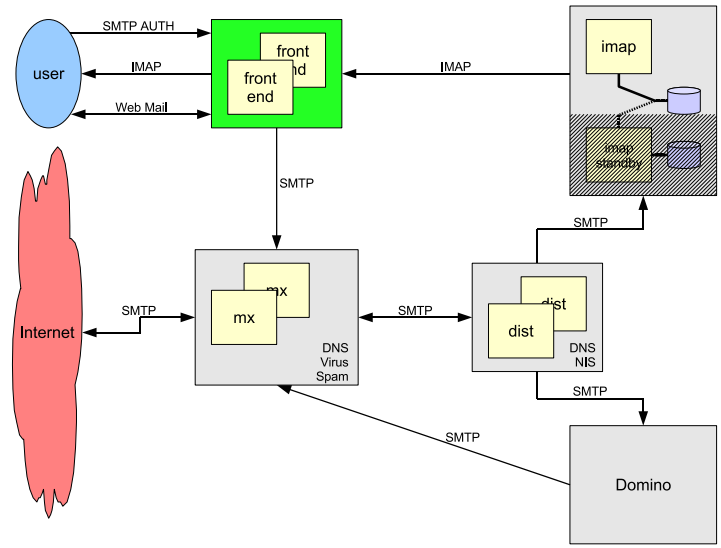
⇒ Sieve-Filtering mit Exim

  - *Direkt ins Maildir*
  - *Logging & Sicherheit*

- „Single Point of Failure”

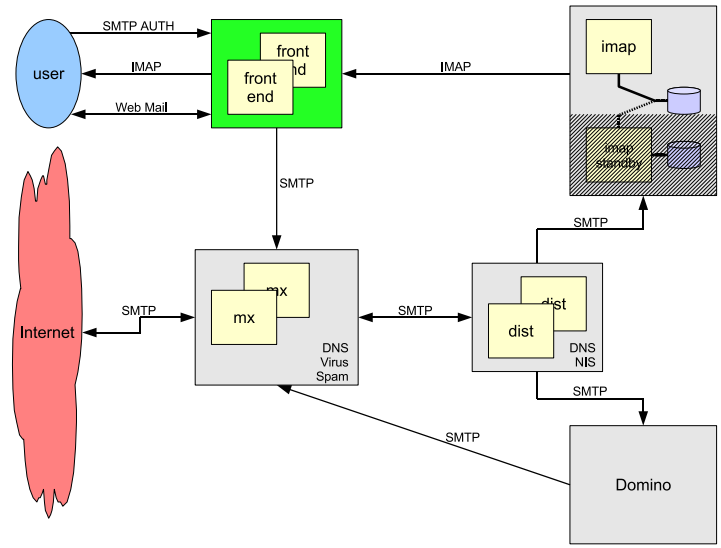
  ⇒ *Shadow-Server + Selbstentwickelte Replikation*

MPI: Funktionseinheit Frontends



- User-Schnittstelle zum Mailsystem
  User-Interaktion getrennt vom Storage-System
- Verschlüsselung mit SSL & TLS
  Max. Client-Unterstützung
- Authentifizierung (SMTP-Auth)
  - *SASL+PAM: Remote-IMAP & Radius*
  - *Files: 2. Fallback mit generischen Usern*

MPI: Funktionseinheit Frontends

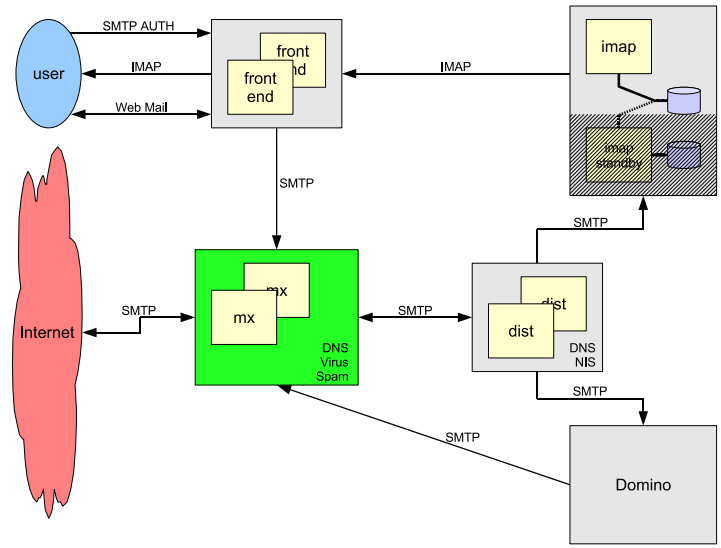


- alternative Ports
  - *High-Ports mit TLS oder SSL*

  $\Rightarrow$ *Max. Erreichbarkeit durch Firewalls*

- Cisco Switch als Load-Balancer
  SMTP- & IMAP-Port redundant ausgelegt

MPI: Feature „Verbose Deny”



- ACL-Erweiterung
  Effizient & Flexibel
- Ablehung zur SMTP-Zeit
  Keine Annahme der Mail ⇒ keine Verantwortlichkeit
  - *Individuelle Ziel-Adresse(n)*
  - *„Nur von außen”*
  - *Ausführliche & individuelle Fehlermeldungen*

MPI: Feature „Verbose Deny” (Beispiele)

```
This message was created automatically by mail delivery software.

A message that you sent could not be delivered to one or more of its
recipients. This is a permanent error. The following address(es) failed:

  peter@mpi-sb.mpg.de
    SMTP error from remote mailer after RCPT TO:<peter@mpi-sb.mpg.de>:
    host interferon.mpi-sb.mpg.de [139.19.1.1]: 550-inexact recipient address:
    550-**************************************************************
    550-*                                                            *
    550-* Sorry, your message to "peter@mpi-sb.mpg.de" was not       *
    550-* delivered.                                                 *
    550-*                                                            *
    550-* There are two computer science faculty members by the name of   *
    550-* "Peter" at Saarbruecken:                                   *
    550-*                                                            *
...
```

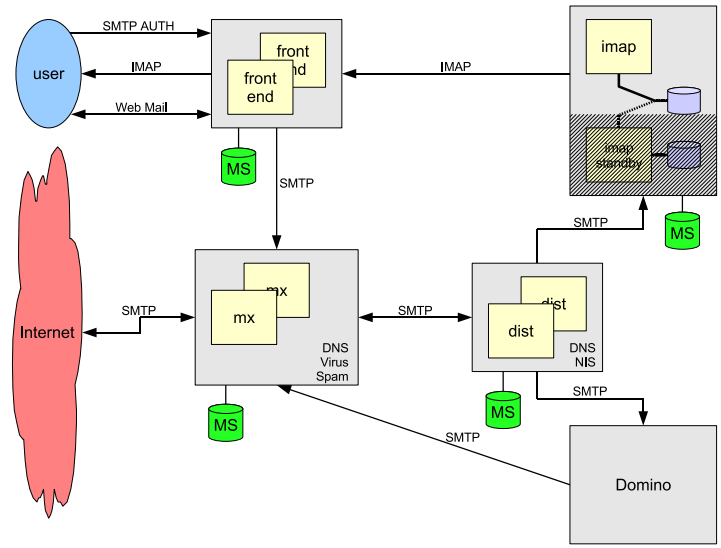
MPI: Feature „Verbose Deny” (Beispiele)

```
This message was created automatically by mail delivery software.

A message that you sent could not be delivered to one or more of its
recipients. This is a permanent error. The following address(es) failed:

  info@vihap3d.org
    SMTP error from remote mailer after RCPT TO:<info@vihap3d.org>:
    host interferon.mpi-sb.mpg.de [139.19.1.1]: 550-address no longer active:
    550-**************************************************************
    550-*                                                            *
    550-* Due to  the large spam  volume, the contact  information for    *
    550-* the ViHAP3D project has  been changed. This email address is    *
    550-* therefore no longer functional. Please see the  project  web    *
    550-* page    http://www.vihap3d.org   for    up-to-date   contact    *
    550-* information.                                                *
    550-* Thanks!                                                     *
    550-* ViHAP3D project team                                        *
    550-*                                                            *
    550 **************************************************************
```

MPI: Feature „Mailstream”



- Backup des „Mail-Streams”

  „Store & Forward” → „Store, Forward, but still store”

- „Kurzzeitige” Speicherung aller empfangenen Mails
  - *Auf jedem Server lokal*
  - ***Sehr viele** Dateien*
- Implementiert als Router und Transport
  - *Erster Router in der Kette, „unseen”*
  - *Transport sorgt für Verteilung in Sub-Directories*

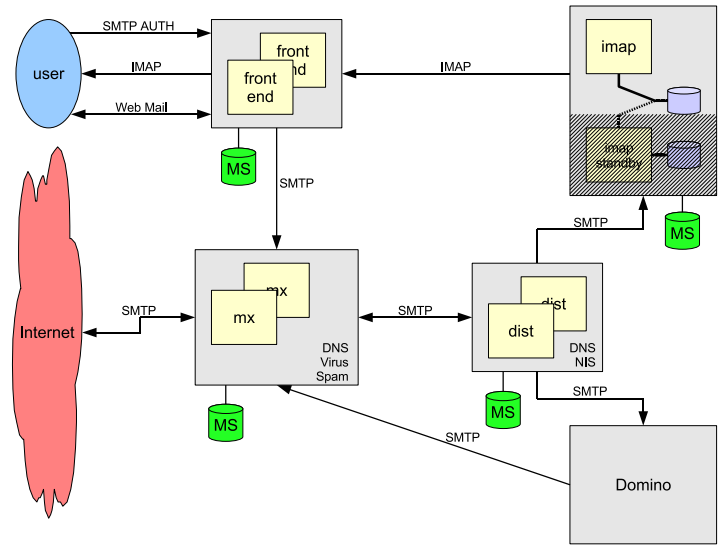
MPI: Feature „Mailstream” (Code)

```
SMTP_PORTS = 25 : 587 : 1025 : 465 : 1465
```

```
Router:
  debug_print = "R: mailstream from \
$sender_host_address to \
$local_part@$domain"
  driver = accept
  unseen = true
  condition = ${if  match_local_part{\
$interface_port}{SMTP_PORTS} {1}{0}}
  no_verify
  no_expn
  transport = mailstream
```

```
Transport:
  debug_print = "T: mailstream from $sender to \
$local_part@$domain"
  driver = appendfile
  directory = /var/spool/mailstream/\
${substr{6}{2}{$tod_logfile}}/\
${substr{11}{2}{$tod_log}}
  create_directory
  delivery_date_add
  envelope_to_add
  return_path_add
  mailstore_format
  directory_mode = 0700
  mode = 0600
  mode_fail_narrower = false
```

MPI: Feature „Mailstream”



- Problem „Privatsphäre” gelöst durch PGP-Verschlüsselung
  - *Out-of-band, per Cron-Job*
  - *Privater Schlüssel sicher verwahrt!*
- Gezieltes *Replay* nach manueller Entschlüsselung
  - *Mit Helfer-Script via Exim-Kommandozeile*
  - *Einfache Realisierung, dank Envelope in separater Datei (unverschlüsselt)*

Szenario MPI: Zusammenfassung

- Redundanz, *falls möglich*
- Hot-Standby für IMAP-Backend
- Entkopplung von sonstiger Infrastruktur

$\Rightarrow$ *Skalierung & Stabilität*

- DMZ & Firewalled Intranet
- verschiedene Backend-Systeme
- Hohe Flexibilität *Auch dank GNU cfengine*

$\Rightarrow$ *Nutzerzufriedenheit*

Szenario MPI: Statistik

- 800 User
  - 350 IMAP-User
  - 150 Notes-User
  - 350 Forward
- ca. 60.000 $\frac{\text{Mails}}{\text{Tag \& Funktionseinheit}}$
- 375GB Storage
  - 100GB IMAP-Server
  - 250GB Domino-Server $(3\times)$
  - 25GB Mailstream Backup (alle 8 Server)

Literatur

- Exim Internet Mailer: http://www.exim.org/
- *Specification of the Exim Mail Transfer Agent*, Version 4.63
  Philip Hazel, 2006, http://exim.org/exim-pdf-current/doc/spec.pdf
- *The Exim SMTP Mail Server – Official Guide for Release 4*
  Philip Hazel, 2003, UIT Cambridge
- *IOS Server Load Balancing Feature in IOS Release 12.2(18)SXE*, Cisco Systems, Inc.
  http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122limit/→
  122sx/12218sxe/slbsxe1.pdf
- Dovecot - Secure IMAP Server: http://www.dovecot.org/
- Perdition: Mail Retrieval Proxy:
  http://www.vergenet.net/linux/perdition/