max planck institut informatik

MAX-PLANCK-GESELLSCHAFT

Max Planck Institute for Software Systems

Exim MTA in der Praxis

Patrick Cernko

pcernko@mpi-sws.org

Max-Planck-Institute
für Softwaresysteme & Informatik
Saarbrücken & Kaiserslautern

24. September 2009

Übersicht

- 1. Teil: Ein paar Grundlagen über Exim
  - Allgemeines, Entstehung, Hintergründe
  - Konzepte von Exim
    - Expansions-Ausdrücke: *Exim's Programmiersprache*
    - ACLs im Detail: „*Wer darf was?*"
- 2. Teil: Praxis
  - Szenario MPI
    Komplexes und doch übersichtliches Mail-System
  - Spezielle Features

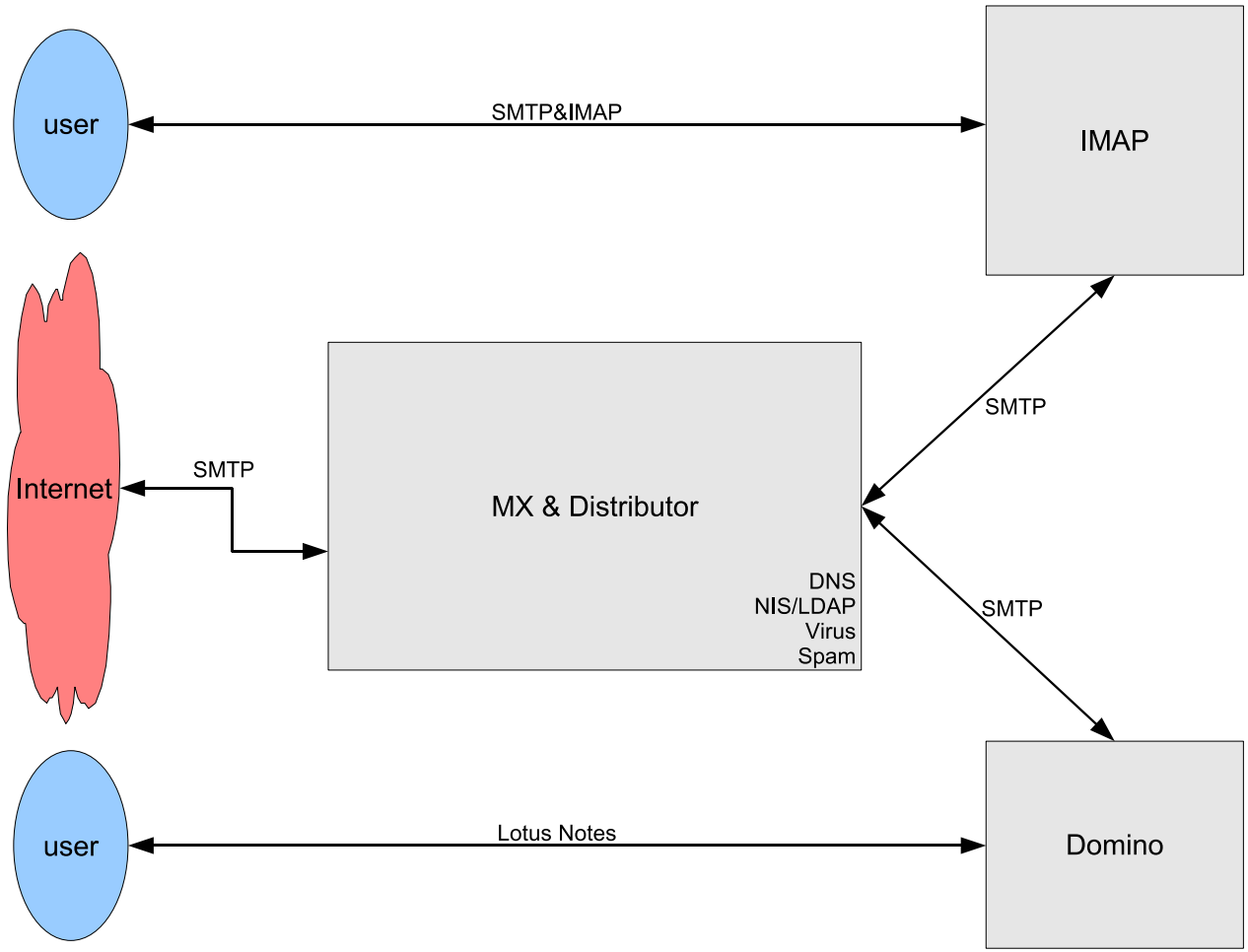
Allgemeines, Entstehung, Hintergründe

- 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: 4.69, OpenSource, GPL
- Distributionen: Debian, Ubuntu, Red Hat, SUSE, Gentoo, FreeBSD, Solaris (CSW)
  Sourcecode portiert auf viele weitere Unix-Derivate

Übersicht (Praxisteil)

- 1. Teil: Grundlagen
- 2. Teil: Praxis
  - Szenario MPI
    Mailserver-Infrastruktur für zwei Forschungsinstitute
    - Blick über das System
      Vom (historischen) Monolith zu den heutigen Funktionseinheiten
    - Spezielle Features
      Außergewöhnliche Anforderungen einfach realisiert

Szenario MPI: Überblick

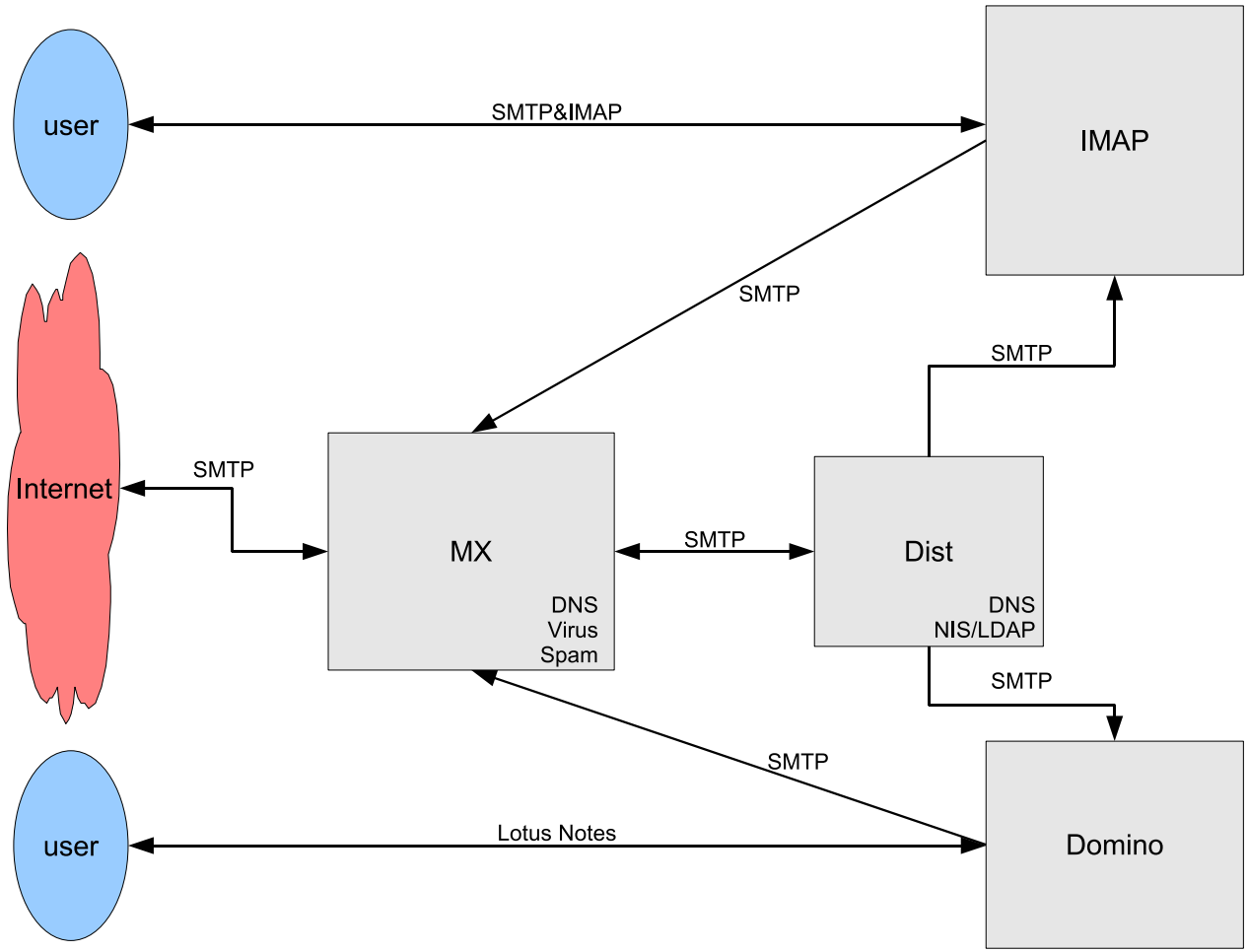


Vogelperspektive

Zwei Backends zur Speicherung

Zentrales Mail-Exchanger-System

Szenario MPI: Überblick

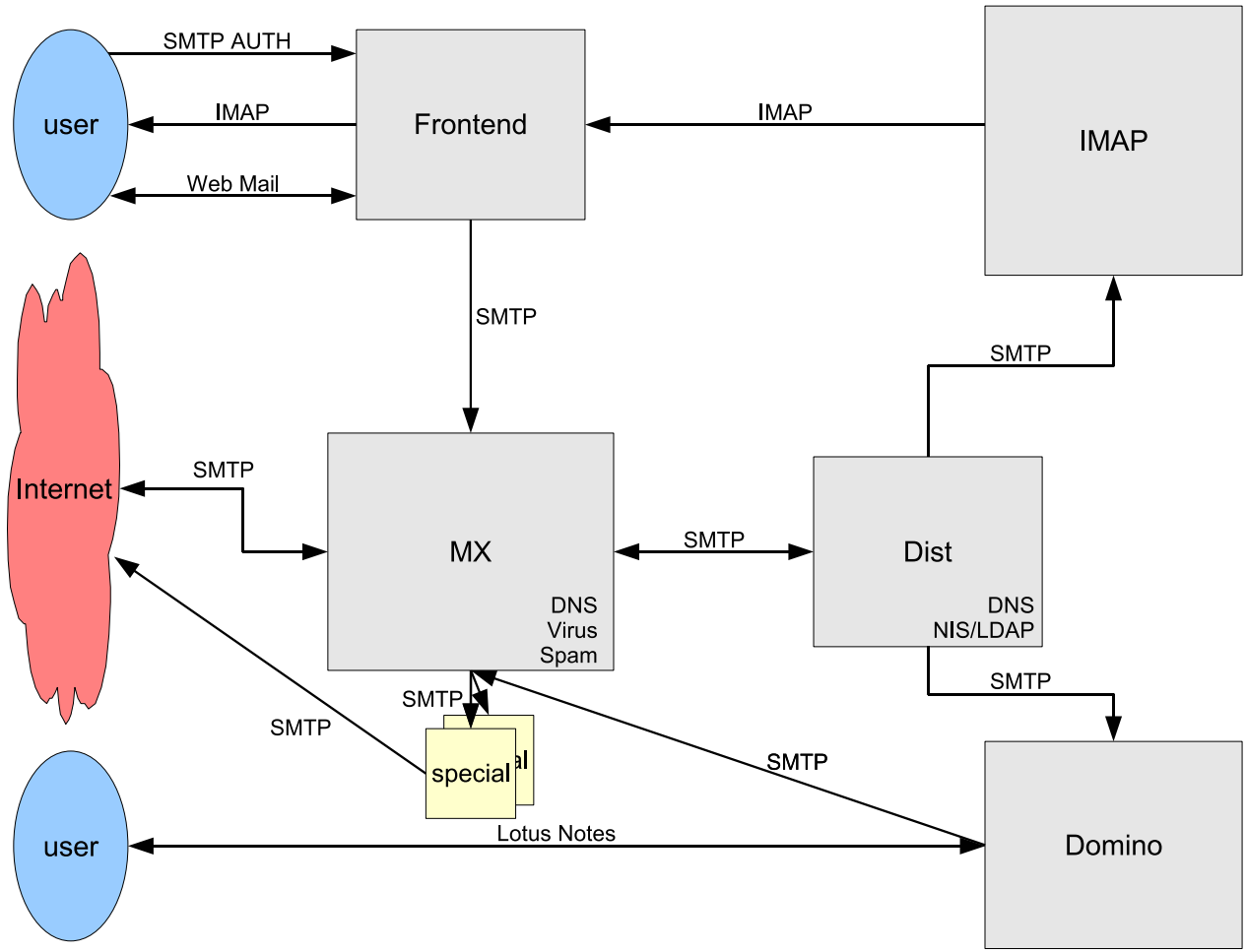


Service-Separation

„Desinfektion” zuerst

Keine Benutzerdaten (NIS) in der DMZ

Szenario MPI: Überblick

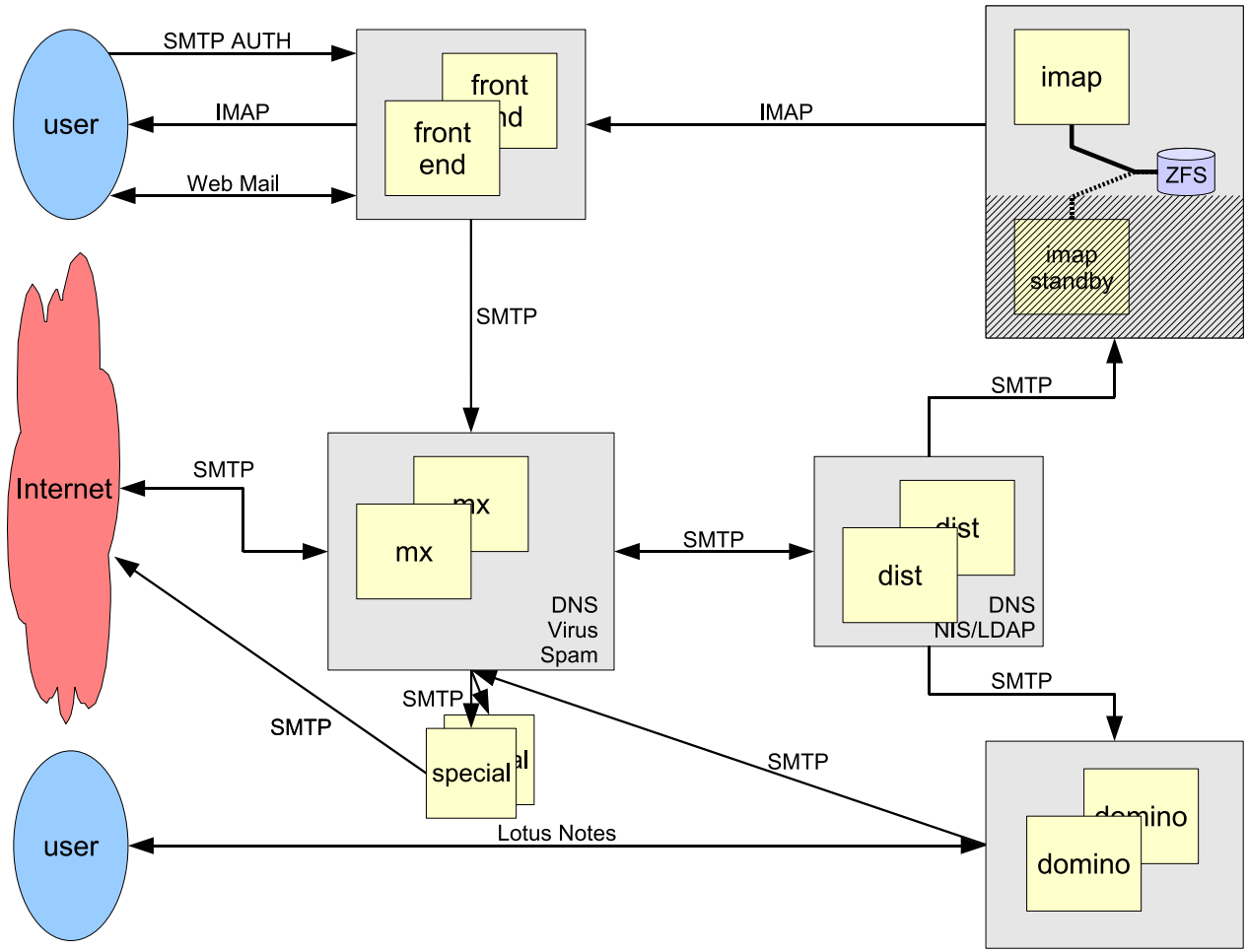


Transparenz & mehr Features

Entkopplung Mail-Storage ↔ User-Interaktion

spezielle Dienste (z.B. Mailman, Alfresco)

Szenario MPI: Überblick

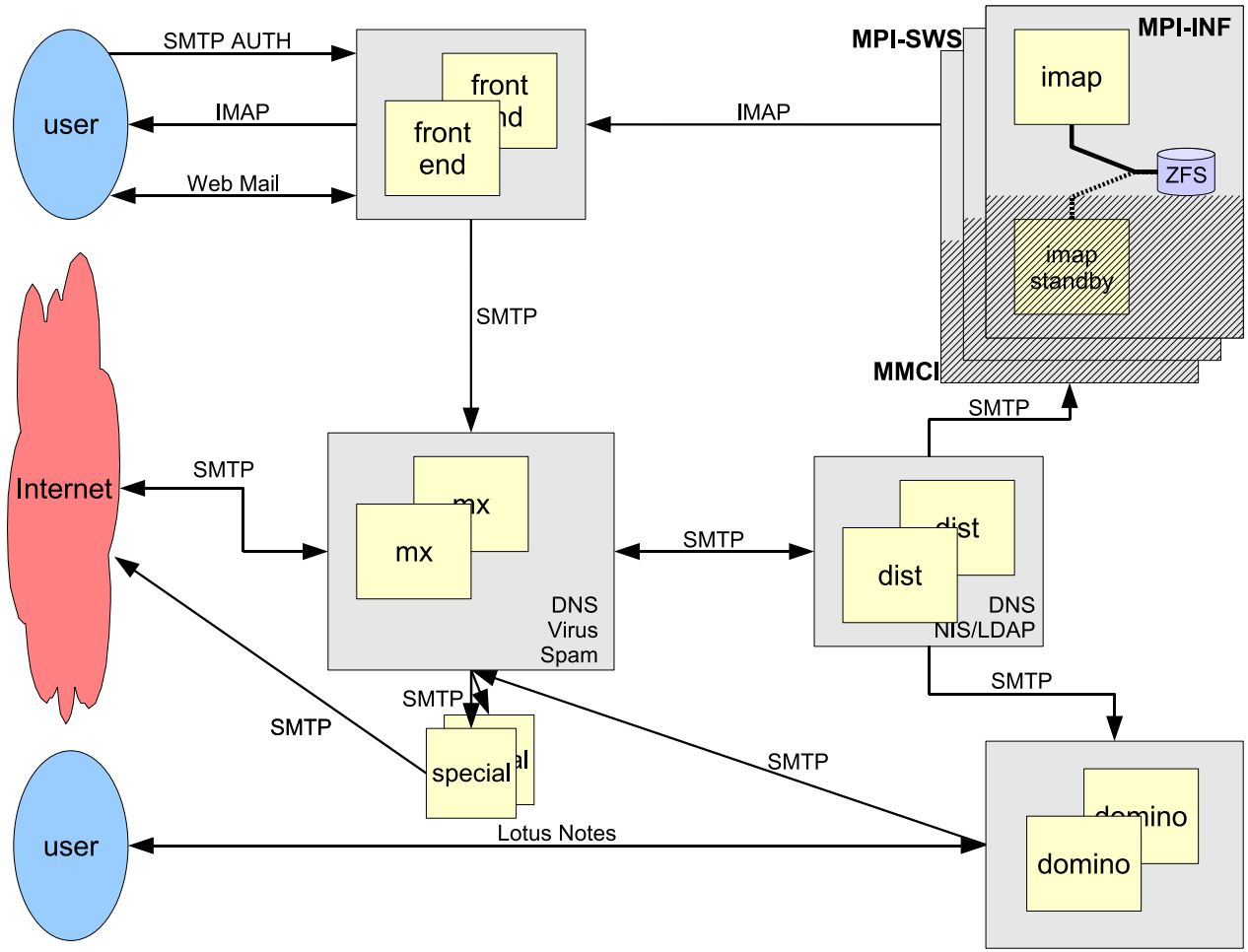


Redundanz & Ausfall-Absicherung

Vollständig Redundante Systeme, falls möglich

Hot-Standby als Alternative

Szenario MPI: Überblick

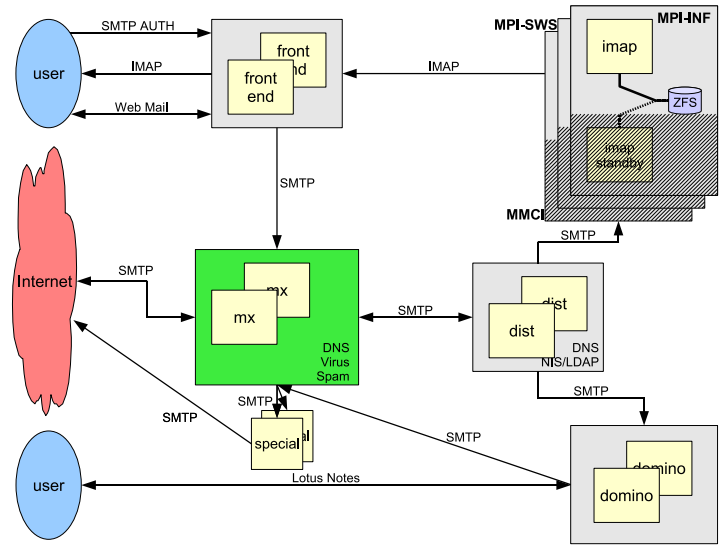


Trennung nach Instituten

dedizierte IMAP-Backends pro Institut/Organisation

⇒ Last-Verteilung & getrennte Authentifizierung möglich

Feature „Verbose Deny”



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

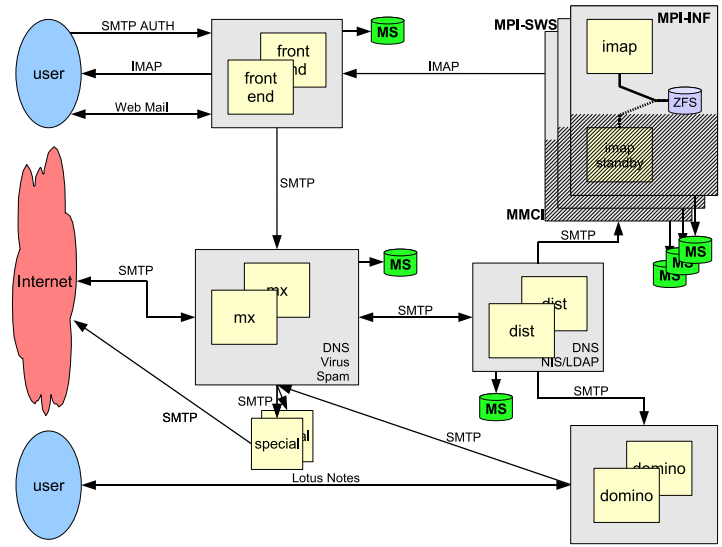
Feature „Verbose Deny” (Beispiel)

```
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 **************************************************************
```

Feature „Mailstream”



- Backup des „Mail-Streams”

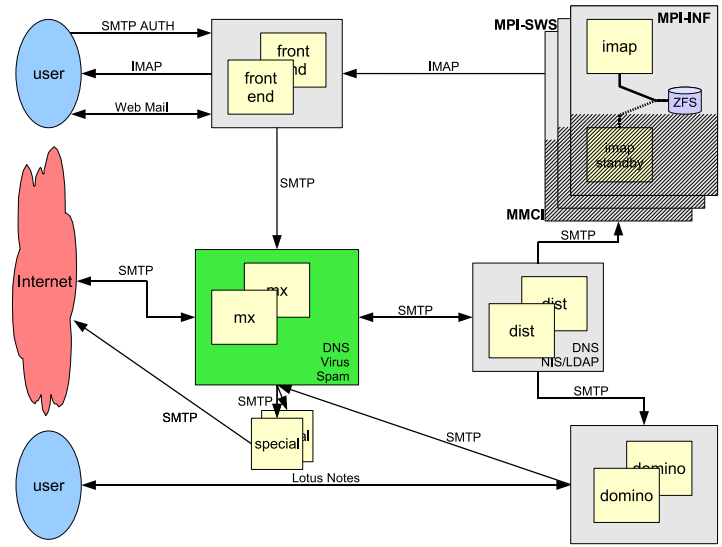
  „Store & Forward” → „Store, Forward, but still store”

- „Kurzzeitige” Speicherung aller empfangenen Mails
- Implementiert als Router und Transport
- Problem „Privatsphäre” gelöst durch PGP-Verschlüsselung
- Gezieltes *Replay* nach manueller Entschlüsselung

Spam-Schutz

- Hohe Anforderungen an Benutzerkontrolle
  - Spam-Ablehnung bei Benutzern umstritten
  - Greylisting vs. automatische Antwortsysteme
- Aber auch Belastung durch Spam
  - Hohes Spam-Aufkommen durch stark publik gemachte Adressen
  - Extreme Spamwellen
  - Backscatter

Feature „Safe-Greylisting”

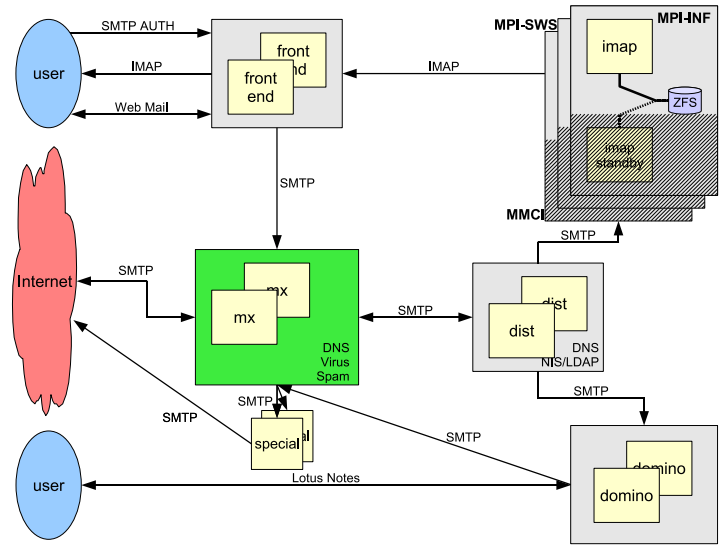


- Blacklisting risikofrei nutzen: *sicheres* Greylisting

  $\Rightarrow$ Nur die *bösen Jungs* müssen warten

- Per Benutzer-Self-Service abschaltbar
- oder: Volles Greylisting bei starkem Spam-Aufkommen einschaltbar

Feature „Spam-Deny”



- Benutzer setzt Spam-Deny-Level

⇒ *eindeutiger* Spam kommt nicht mehr ins System

⇒ Bei mehreren Empfängern entscheidet Maximum

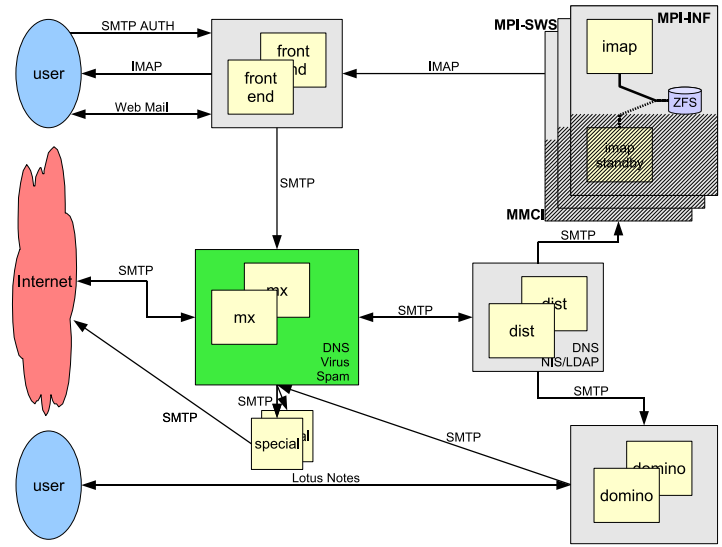
⇒ Benachrichtigung von False-Positives beim Sender

- Mails unter dem Level werden immer noch markiert

⇒ Rest-Spam wird aussortiert oder manuell geprüft

⇒ Entlastung für Nutzer und Backup

Feature „Bounce-Protection”



- Schutz vor Backscattern durch *Absendermarkierung*
- Nach Aktivierung via Benutzer-Self-Service:
  - ⇒ Ausgehende Mails werden markiert
  - ⇒ Rückläufern ohne Markierung werden mit hohem Spam-Level klassifiziert

Übersicht (Theorieteil)

- 1. Teil: Ein paar Grundlagen über Exim
  - Allgemeines, Entstehung, Hintergründe
  - Konzepte von Exim
    - Expansions-Ausdrücke: *Exim's Programmiersprache*
    - ACLs im Detail: *„Wer darf was?"*
- 2. Teil: Praxis
  - Szenario MPI
    Komplexes und doch übersichtliches Mail-System
  - Spezielle Features

Exim-Konzepte: Expansions-Ausdrücke (Beispiel)

```
tls_certificate = /etc/ssl/mail.mydomain.de.pem
```

- *Laut Dokumentation von* `tls_certificate` *auf den ersten Blick nur eine Datei möglich*

Exim-Konzepte: Expansions-Ausdrücke

- „Programmiersprache” von Exim

Exim-Konzepte: Expansions-Ausdrücke

- „Programmiersprache” von Exim
- Großes Set an Funktionen & Variablen

```
$authenticated_id, $load_average, $smtp_count_at_connection_start, $spool_space,

${substr{6}{2}{$tod_logfile}}
```

  - *PCRE Library*

    ```
    ${sg {$spam_bar} {\N\+\N} {X}}
    ```

  - 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-Konzepte: Expansions-Ausdrücke

- „Programmiersprache” von Exim
- Großes Set an Funktionen & Variablen

```
$authenticated_id, $load_average, $smtp_count_at_connection_start, $spool_space,

${substr{6}{2}{$tod_logfile}}
```

  - *PCRE Library*

    ```
    ${sg {$spam_bar} {\N\+\N} {X}}
    ```

  - 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-Konzepte: Expansions-Ausdrücke (Beispiel)

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

- *Laut Dokumentation von* `tls_certificate` *scheinbar nur eine Datei möglich*
- *dank Expansions-Ausdruck jedoch dynamisierbar*
- *Statt* `fail` *auch einfach ein weiterer geschachtelter Ausdruck*

Exim-Konzepte: ACLs

- „Access Control Lists” für (SMTP-)Befehle

  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-Konzepte: ACLs

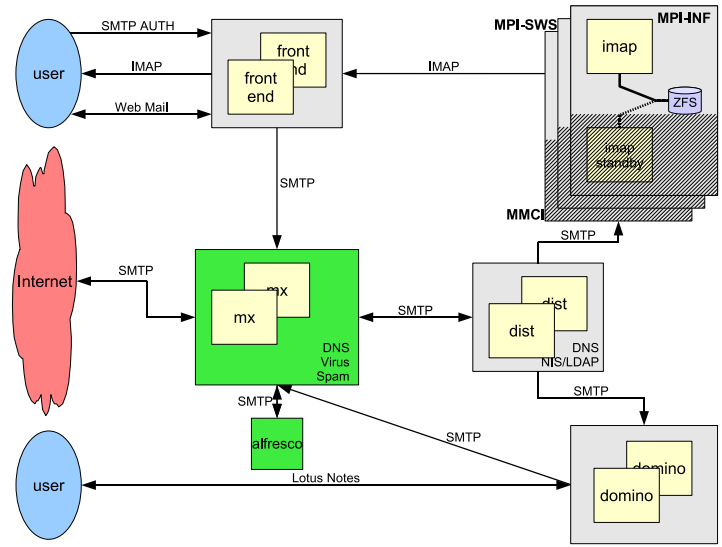
- Nicht nur Ablehnung!
  Logging, Header-Ergänzung, „State-Control”, temp. Ablehnung, Blackhole, gesteuerte Delays
- Spam- & Virenfilter
  - *Einbindbar über fertige Primitive oder vorbereitete Sockets*

  $\Rightarrow$ *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.

Integration von spezialisierten Diensten: Alfresco



- Mail-In Feature von Alfresco: *Dokumente per Mail aktualisieren*
- Problem 1: Empfänger dynamisch, bei MX unbekannt

$\Rightarrow$ Callout-Verification zur SMTP-Zeit beim dahinter stehendend System

- Problem 2: Alfresco lehnt falsche Absender/Empfänger erst nach `DATA` ab

$\Rightarrow$ Lokaler Exim vor Alfresco, prüft Absender im LDAP

Danke!

Literatur

- *Specification of the Exim Mail Transfer Agent*, Version 4.68
  Philip Hazel, 2007, 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/
- *Using Postgrey with Exim*, Guy Antony Halse, 2007, Postgrey distribution,
  http://postgrey.schweikert.ch/
- *Tweak your MTA: Spamschutz mit Tricks*, Tobias Eggendorfer, 2007, 3. Mailserver-Konferenz
- *Bounce Address Tag Validation (BATV)*, Network Working Group, 2008,
  http://mipassoc.org/batv/draft-levine-smtp-batv-01.html