

Ideen für grosse Mailstores

Mailbox Cluster Konzepte

Felix J. Ogris (fjo@dts.de)

Version 1.0
2008-05-25

Inhaltsverzeichnis

1 Schema	4
2 Storage	5
2.1 Mailbox- und Backupserver mit lokalem Speicher	5
2.2 Mailboxserver mit Speicher im SAN	6
2.3 Mailboxserver mit Speicher auf Storagebox	7
2.4 Mailbox- und Backupserver mit Speicher auf eigener Storagebox	8
2.5 Mailbox- und Backupserver mit Speicher in redundantem SAN	9
3 Distribution	11
3.1 Cyrus Murder mit Loadbalancer	11
3.2 Cyrus Murder im Active-Active Cluster	12
3.3 Perdition mit Loadbalancer	13
3.4 Perdition im Active-Active Cluster	14
4 Access	15
4.1 Webmail	15
4.1.1 Webmail ohne Adressbücher und Benutzereinstellungen	15
4.1.2 Webmail mit zentral vorgehaltenen Adressbüchern und Benutzereinstellungen	17
4.1.3 Webmail mit Anmeldeproxy und verteilten Adressbüchern und Benutzereinstellungen	18
4.2 Mailinjection	20
5 Sinnvolles Konzept	22
A Quellen	24

Abbildungsverzeichnis

1	Schematische Darstellung	4
2	Mailboxserver mit lokalem Speicher, Synchronisation auf identische Backupserver per Cyrus-Syncclient	5
3	Mailboxserver mit Speicher im SAN unter Einsatz eines Cluster Dateisystems, Synchronisation per Hardware	6
4	Mailboxserver mit Speicher auf Storagebox, Synchronisation per Hardware	7
5	Mailbox- und Backupserver mit Speicher auf Storagebox, Synchronisation per Cyrus-Syncclient	8
6	Mailbox- und Backupserver mit Speicher in redundantem SAN, Synchronisation per Hardware	9
7	Cyrus Murder mit Mupdate Server und Loadbalancer	11
8	Cyrus Murder mit Mupdate Server und Active-Active Cluster	12
9	Perdition mit Loadbalancer	13
10	Perdition im Active-Active Cluster	14
11	Einfacher Webmail Cluster mit Loadbalancer	15
12	Einfacher Webmail Cluster im Active-Active Betrieb	16
13	Webmail Cluster mit Loadbalancer und zentraler Datenbank mit Backup	17
14	Webmail Cluster im Active-Active Betrieb mit zentraler Datenbank samt Backup	17
15	Webmail Cluster mit Loadbalancer und verteilten Daten samt Backup	18
16	Webmail Cluster im Active-Active Betrieb mit verteilten Daten samt Backup	18
17	SMTP-Relay mit Loadbalancer	20
18	SMTP-Relay als Active-Active Cluster	21
19	Gefaltetes Setup	22

1 Schema

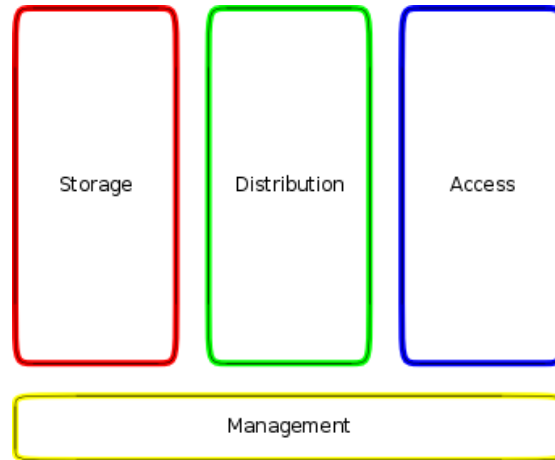


Abbildung 1: Schematische Darstellung

Klassisches 3-Tier-Problem:

Storage beschreibt, wie und wo Mailboxen gespeichert werden

Distribution beschreibt, wie die Mailbox eines bestimmten Benutzers gefunden wird

Access beschreibt, wie Benutzer und andere Systeme mit dem Mailcluster kommunizieren.

Ferner muss ein übergreifendes Management existieren, welches Stammdaten wie Benutzernamen und Passwörter in die jeweilige Schicht verteilt und Betriebsdaten wie Loginstatistiken akkumuliert und aufbereitet.

Innerhalb jeder der drei bzw. 4 Schichten müssen folgende Punkte gewährleistet sein:

- Ausfallsicherheit der Schicht als ganzes und jeder einzelnen Komponente, insbesondere der Schnittstellen
- Lastverteilung (Vermeidung von Hot Spots)
- Einfaches Erweitern bei Änderung des Lastverhaltens (horizontale Skalierbarkeit)

2 Storage

2.1 Mailbox- und Backupserver mit lokalem Speicher

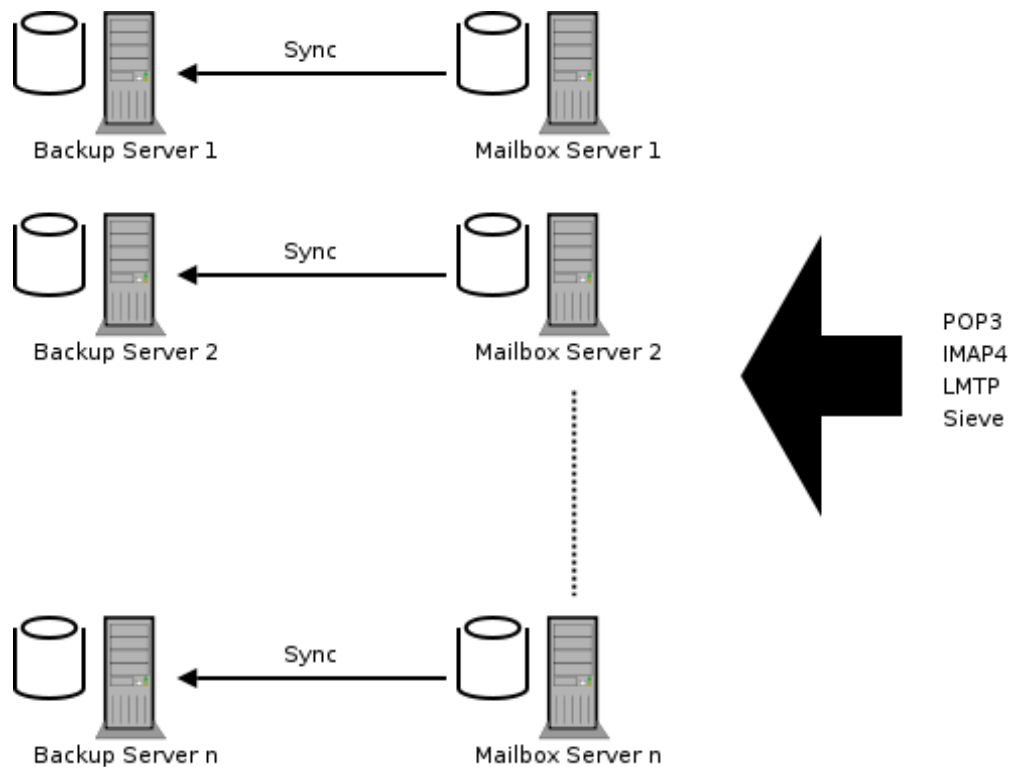


Abbildung 2: Mailboxserver mit lokalem Speicher, Synchronisation auf identische Backupserver per Cyrus-Syncclient

Vorteile:

- Ausfallsicherheit durch Backupserver
- leicht erweiterbar
- Lastverteilung möglich (s.u.)

Nachteile:

- Cyrus-Syncclient noch relativ neu
- Synchronisation arbeitet verzögert
- Lastverteilung abhängig vom Managementlayer (Verteilung der Mailboxen im Round-Robin-Verfahren o.ä. auf die einzelnen Server)
- manuelles Umschalten im Fehlerfall

2.2 Mailboxserver mit Speicher im SAN

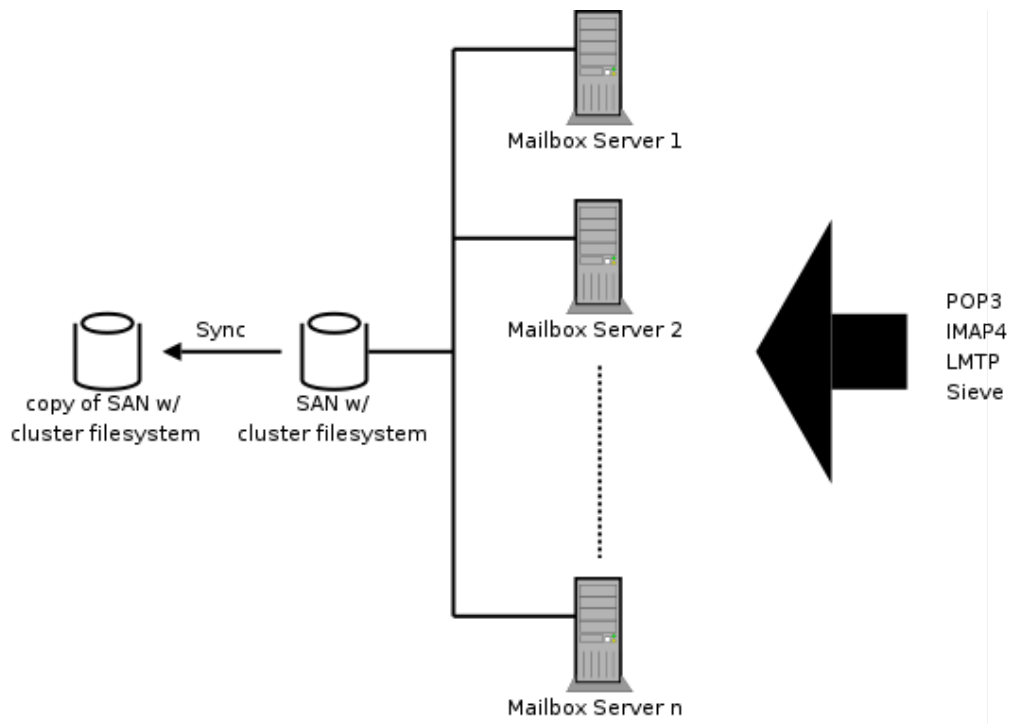


Abbildung 3: Mailboxserver mit Speicher im SAN unter Einsatz eines Cluster Dateisystems, Synchronisation per Hardware

Vorteile:

- synchrone SAN-Kopie
- Lastverteilung über die Mailboxserver möglich

Nachteile:

- keine Ausfallsicherheit eines Mailboxservers
- SAN wird zum Engpass bei grosser Serveranzahl
- schlecht erweiterbar, zudem Investitionssprung bei Anschaffung eines weiteren SAN-Systems
- erfordert Cyrus Murder
- noch sehr neues Feature im Cyrus Murder

2.3 Mailboxserver mit Speicher auf Storagebox

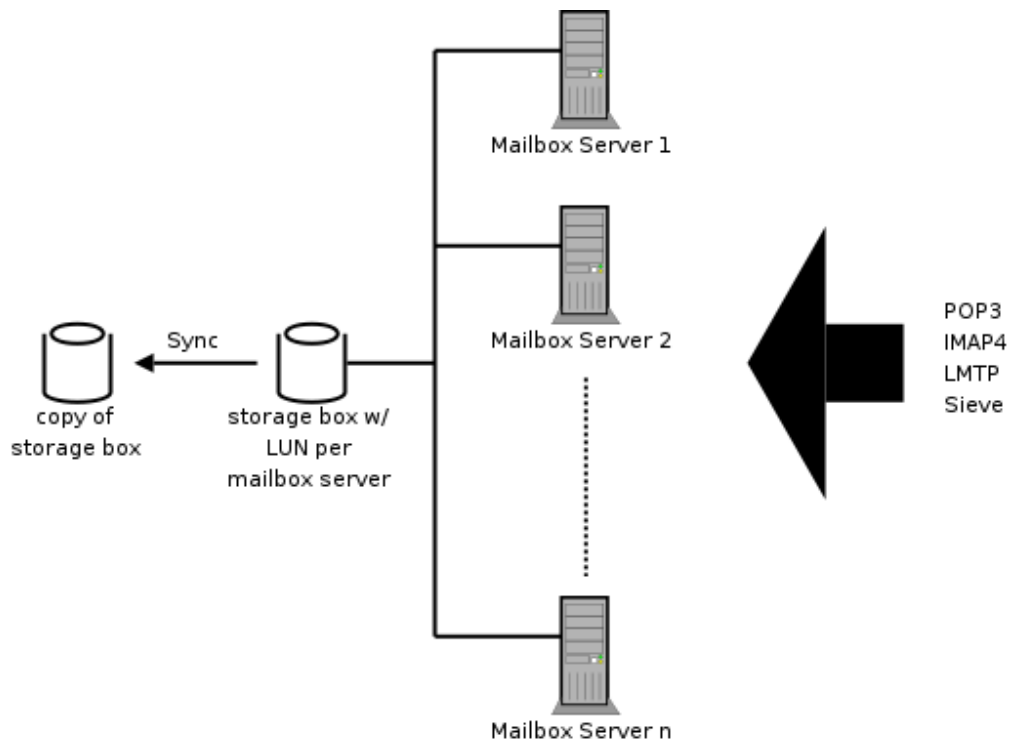


Abbildung 4: Mailboxserver mit Speicher auf Storagebox, Synchronisation per Hardware

Vorteile:

- synchrone SAN-Kopie
- Lastverteilung über die Mailboxserver möglich

Nachteile:

- keine Ausfallsicherheit eines Mailboxservers
- SAN wird zum Engpass bei grosser Serveranzahl
- schlecht erweiterbar, zudem Investitionssprung bei Anschaffung eines weiteren SAN-Systems

2.4 Mailbox- und Backupserver mit Speicher auf eigener Storagebox

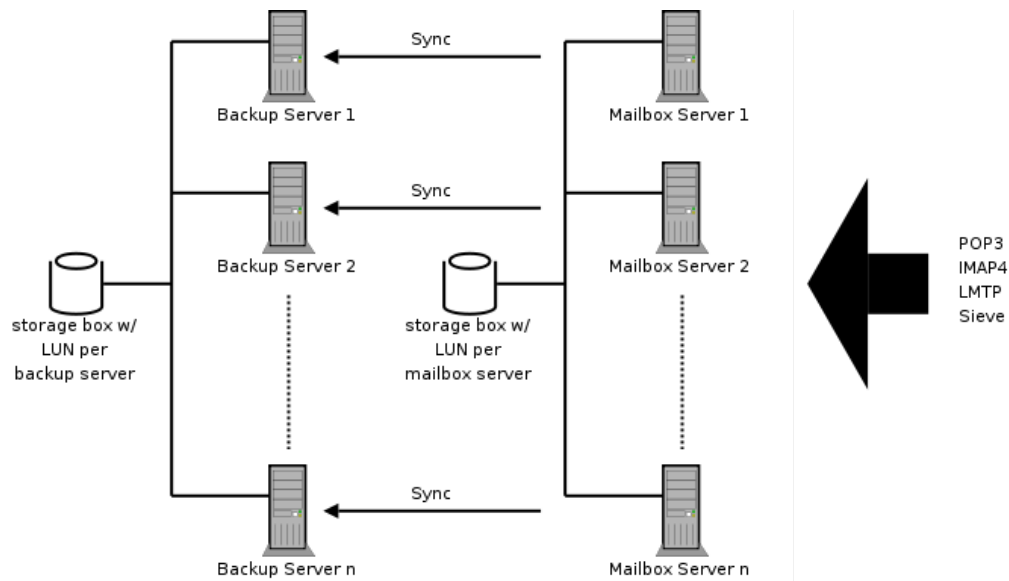


Abbildung 5: Mailbox- und Backupserver mit Speicher auf Storagebox, Synchronisation per Cyrus-Syncclient

Vorteile:

- Ausfallsicherheit durch Backupserver
- Lastverteilung über die Mailboxserver möglich

Nachteile:

- Cyrus-Syncclient noch relativ neu
- Synchronisation arbeitet verzögert
- SAN wird zum Engpass bei grosser Serveranzahl
- schlecht erweiterbar, zudem Investitionssprung bei Anschaffung eines weiteren SAN-Systems
- manuelles Umschalten im Fehlerfall

2.5 Mailbox- und Backupserver mit Speicher in redundantem SAN

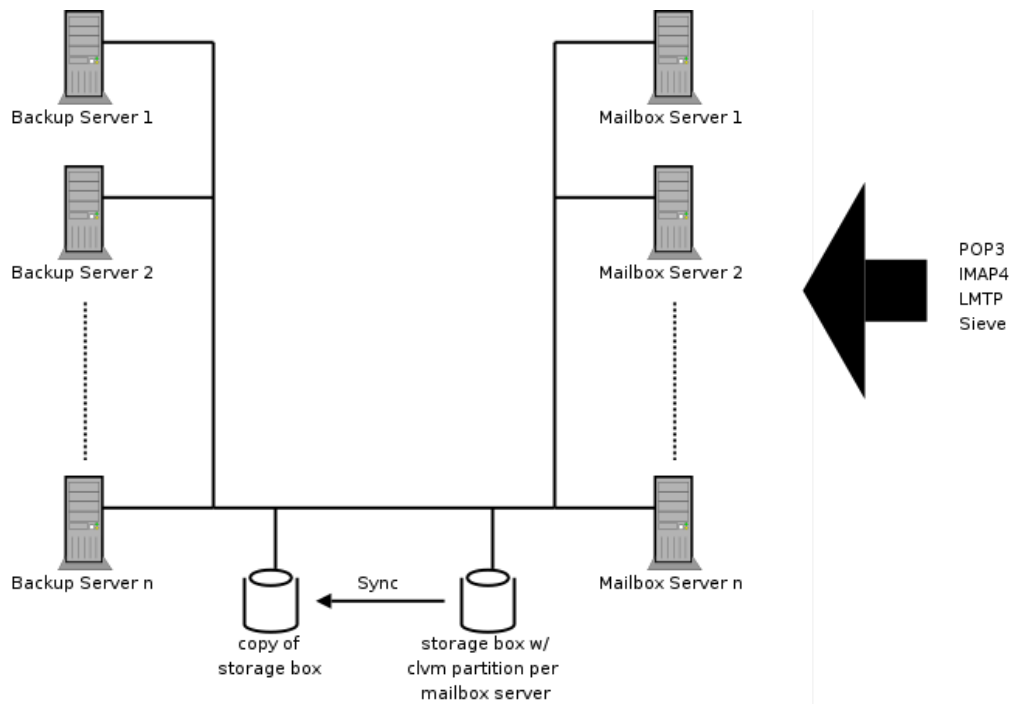


Abbildung 6: Mailbox- und Backupserver mit Speicher in redundantem SAN, Synchronisation per Hardware

Idee:

- Storageboxen in sich ausfallsicher, 2-PC-Synchronisation in Hardware
- jeder Server hängt an beiden Storageboxen
- pro Mailbox-/Backupserverpaar je eine identische clvm-Partition auf beiden Storageboxen
- diese beiden Partitionen, die auf den zwei Storageboxen liegen, erscheinen nach aussen mit der gleichen WWN, so dass jeder Server denkt, er hätte zwei Wege zu nur einer Partition
- per Multipathing wird immer nur auf einen Pfad und somit eine Partition geschrieben
- stirbt eine Storagebox, sorgt das Multipathing dafür, dass auf die verbleibende Box geschrieben wird
- stirbt der Mailboxserver, sorgt das Fencing vom clvm dafür, dass nur noch der Backupserver die Storageboxen sieht (Fencing per Brocade-Switch)

Vorteile:

- Ausfallsicherheit durch Backupserver
- Lastverteilung über die Mailboxserver möglich

- automatisches Failover

Nachteile:

- SAN wird zum Engpass bei grosser Serveranzahl
- schlecht erweiterbar, zudem Investitionssprung bei Anschaffung eines weiteren SAN-Systems
- ungetestet, geht das überhaupt?

3 Distribution

3.1 Cyrus Murder mit Loadbalancer

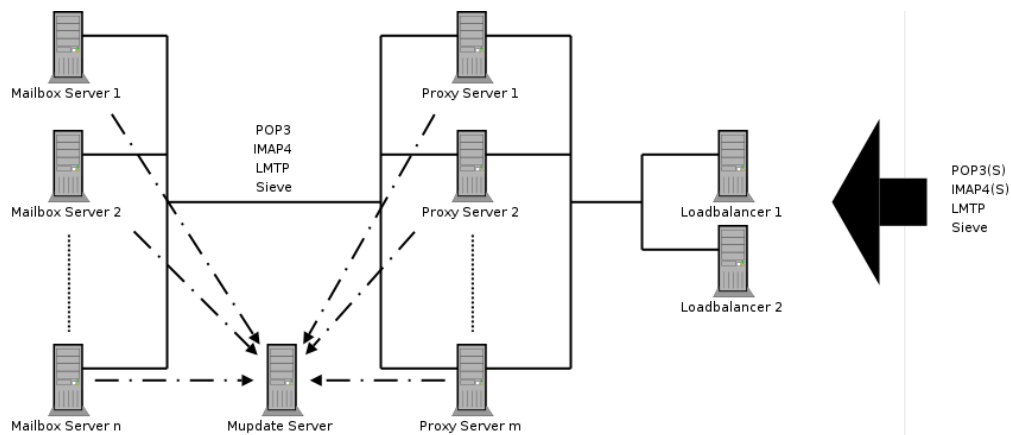


Abbildung 7: Cyrus Murder mit Mupdate Server und Loadbalancer

Idee:

- Jeder Mailboxserver teilt dem Mupdate Server mit, welche Mailboxen er kennt
- Proxyserver fragen beim Login eines Benutzers den Mupdate Server, auf welchen Backendserver die Mailbox liegt
- HA-Loadbalancer mit keepalived zur Lastverteilung über die Proxys
- Proxys terminieren SSL Verbindungen

Vorteile:

- ausfallsichere Proxyserver
- Lastverteilung
- einfach zu erweitern
- Änderungen in der Mailboxstruktur sind sofort auf den Proxys sichtbar

Nachteile:

- Mupdate Server ist single point of failure; evtl. lösbar durch zwei Mupdate Knoten im Active-Passive Cluster mit Perl-/SSH-Skripten, die die Mailboxserver im Fehlerfall zum Resync auffordern

3.2 Cyrus Murder im Active-Active Cluster

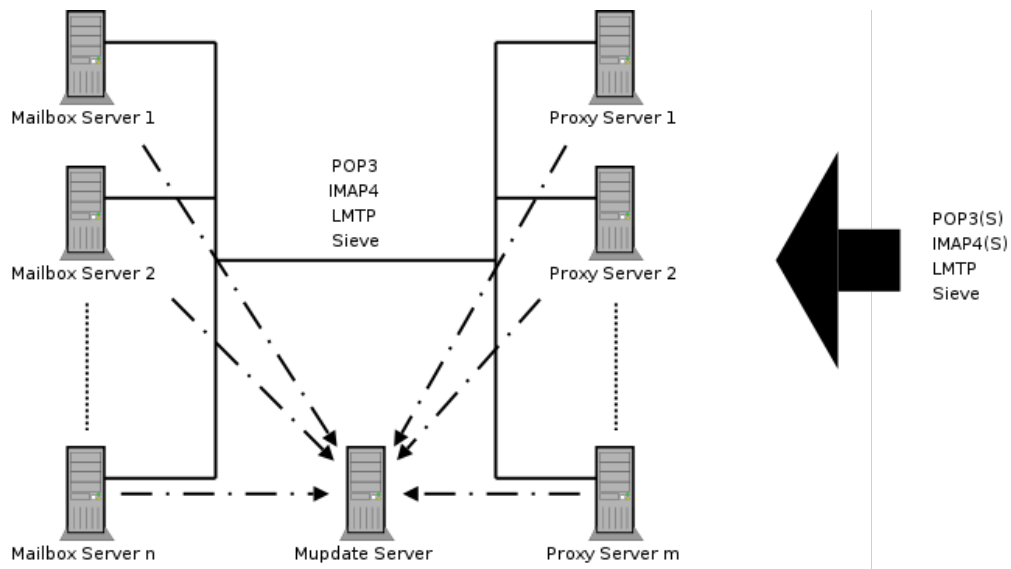


Abbildung 8: Cyrus Murder mit Mupdate Server und Active-Active Cluster

Idee:

- Proxyserver bilden einen Cluster mit virtueller IP
- Anfragen verteilt der Cluster intern selbstständig
- alle Clusterknoten müssen sich per Multicast o.ä. sehen

Vorteile gegenüber 3.1:

- spart 2 Loadbalancer

Nachteile gegenüber 3.1:

- FreeBSD: Kernelpatch vom Autor notwendig, kein Test in freier Wildbahn
- Switches mit Arp-Snooping sind ungeeignet

3.3 Perdition mit Loadbalancer

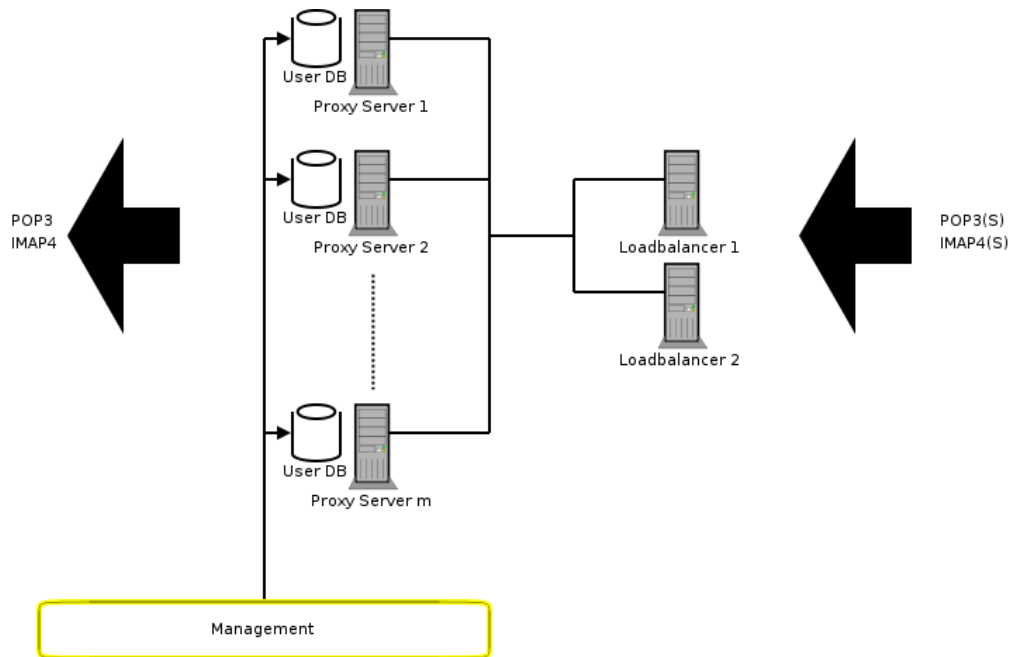


Abbildung 9: Perdition mit Loadbalancer

Idee:

- Generischer Mailproxy, z.B. Perdition, mit lokaler Datenbank, die Benutzernamen zu Mailboxservern mappt
- HA-Loadbalancer mit keepalived zur Lastverteilung über die Proxys
- Proxys terminieren SSL Verbindungen

Vorteile:

- ausfallsichere Proxyserver
- Lastverteilung
- einfach zu erweitern
- Migration von bestehenden Mailservern ohne Unterbrechung des Betriebs möglich

Nachteile:

- Perdition spricht nur IMAP(s) und POP3(S), kein LMTP, kein SMTP, kein Sieve
- lokale Datenbanken werden zeitverzögert vom Managementlayer synchronisiert

3.4 Perdition im Active-Active Cluster

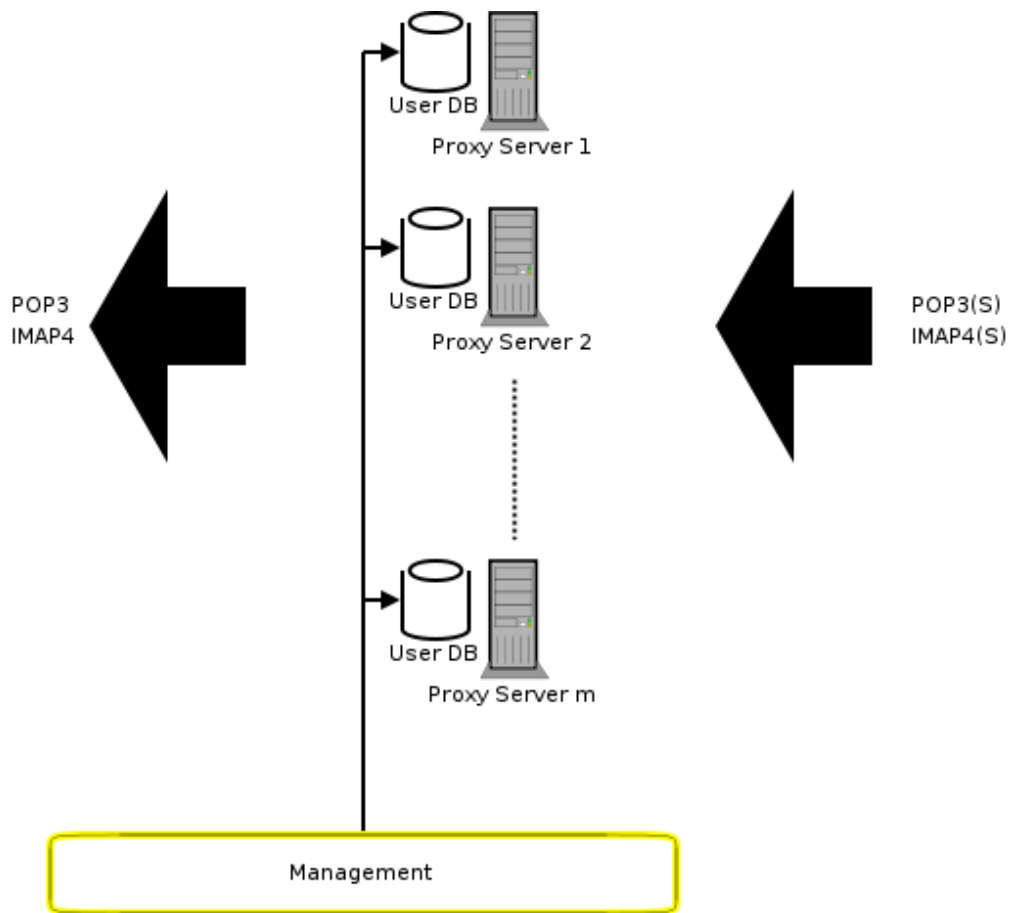


Abbildung 10: Perdition im Active-Active Cluster

Kombination der Ideen, Vor- und Nachteile von [3.3](#) und [3.2](#)

4 Access

Da der Distributionlayer den Zugang per IMAP4(S) und POP3(S) bereitstellt, muss nur noch der Zugang per HTTP(S) (Webmail) und LMTP/SMTP abgebildet werden.

4.1 Webmail

4.1.1 Webmail ohne Adressbücher und Benutzereinstellungen

Triviales Setup: Webmailserver dienen als HTTP(S)-IMAP-Gateway, Aufbau mit Loadbalancer (Abbildung 11) oder als Active-Active Cluster (Abbildung 12).

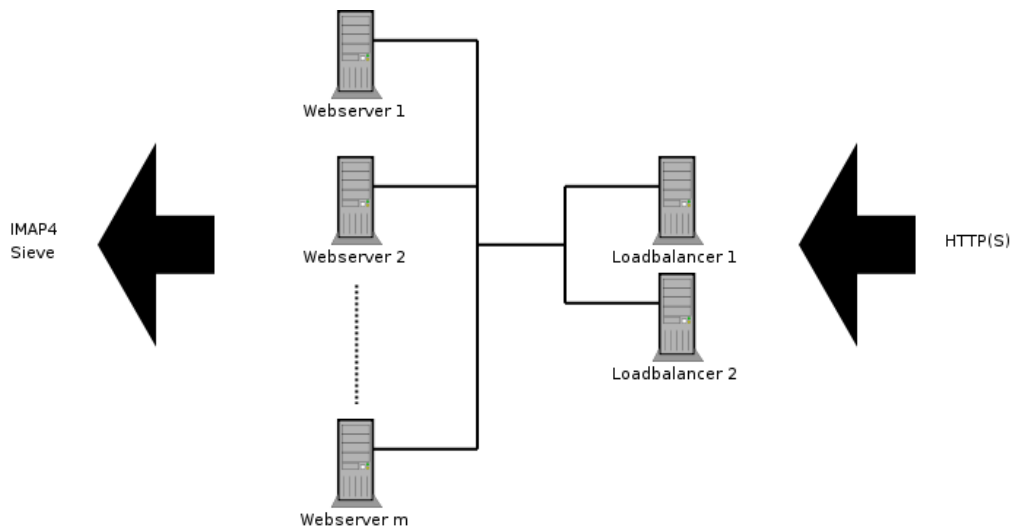


Abbildung 11: Einfacher Webmail Cluster mit Loadbalancer

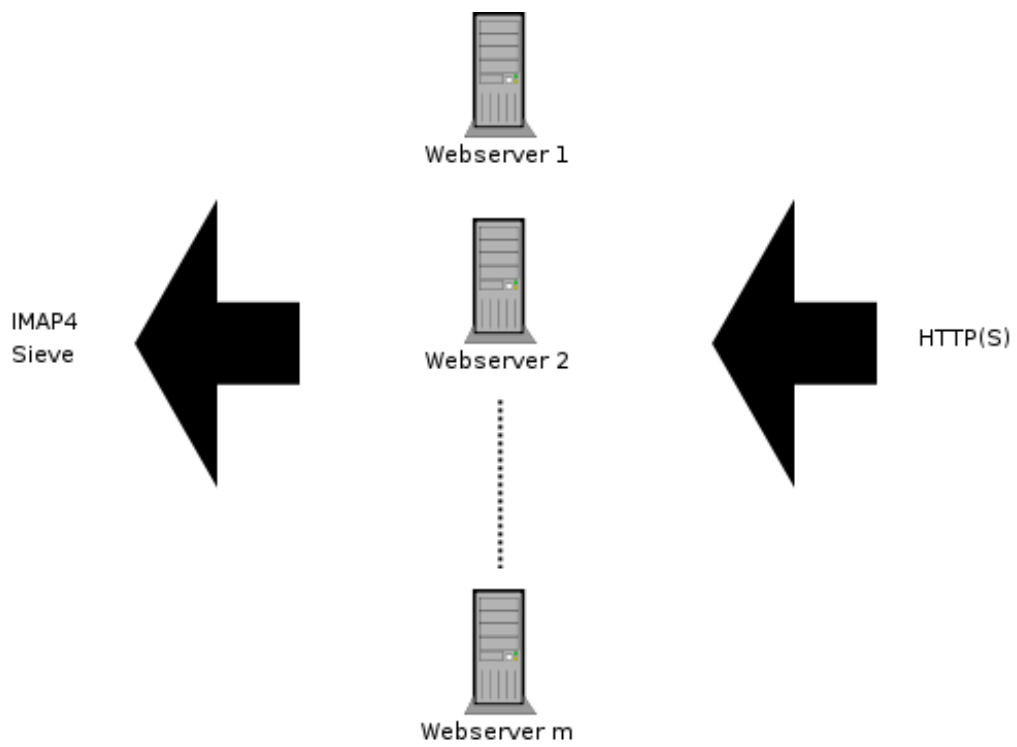


Abbildung 12: Einfacher Webmail Cluster im Active-Active Betrieb

4.1.2 Webmail mit zentral vorgehaltenen Adressbüchern und Benutzereinstellungen

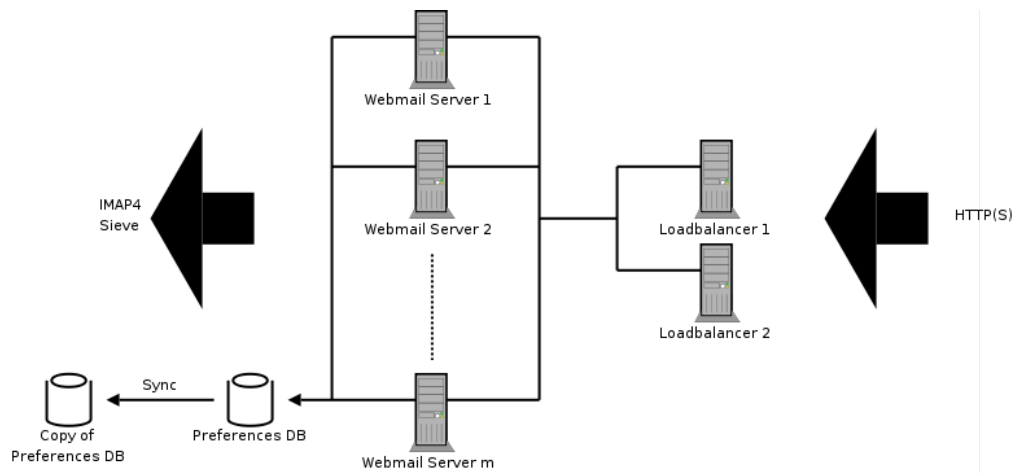


Abbildung 13: Webmail Cluster mit Loadbalancer und zentraler Datenbank mit Backup

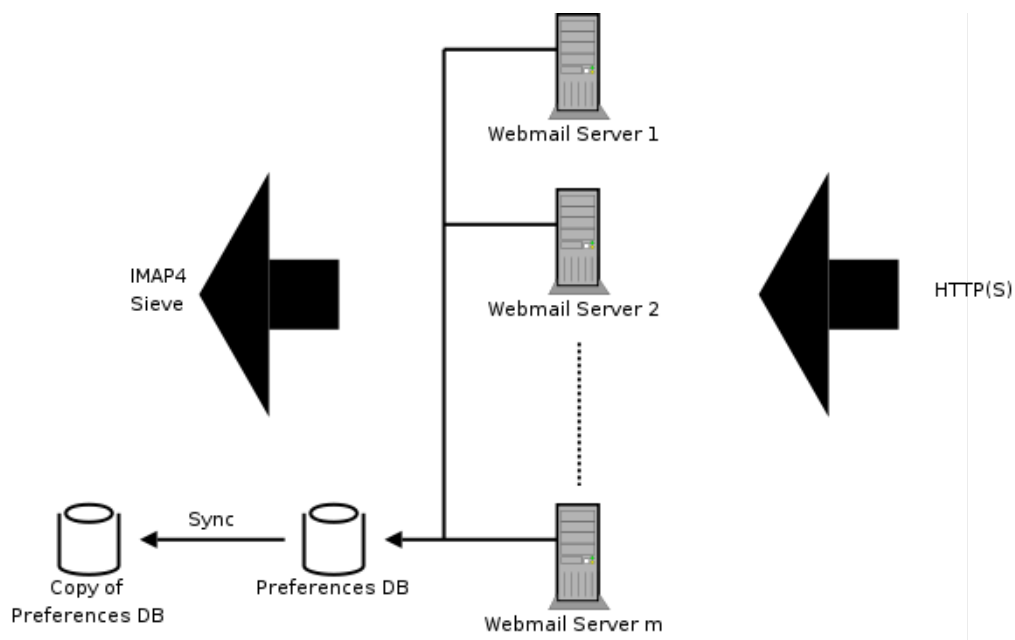


Abbildung 14: Webmail Cluster im Active-Active Betrieb mit zentraler Datenbank samt Backup

Bei Verwendung von Squirrel muss die zentrale Datenbank durch ein SAN-Device & -Backup mit Clusterfilesystem ersetzt werden.

Nachteile:

- zentrale Datenbank wird zum Flaschenhals

4.1.3 Webmail mit Anmeldeproxy und verteilten Adressbüchern und Benutzereinstellungen

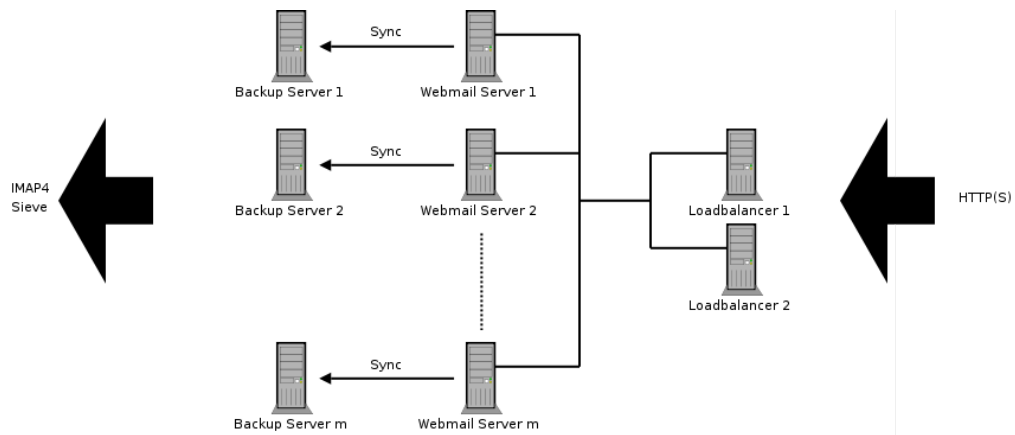


Abbildung 15: Webmail Cluster mit Loadbalancer und verteilten Daten samt Backup

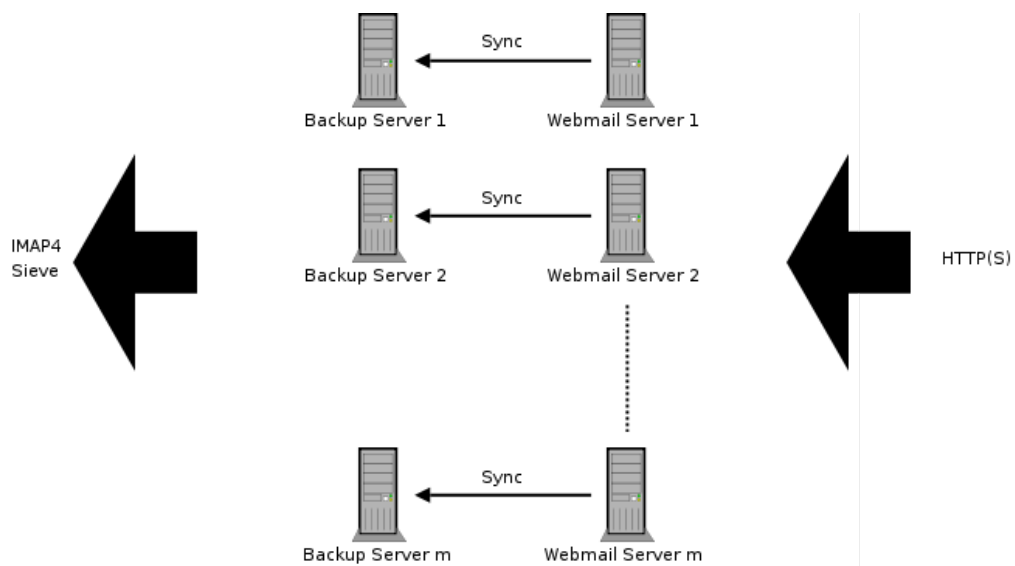


Abbildung 16: Webmail Cluster im Active-Active Betrieb mit verteilten Daten samt Backup

Idee:

- Benutzer melden sich an einem vorgeschaltetem Proxyserver an
- anhand des Benutzernamens wird auf den eigentlichen Webmailserver weitergeleitet
- Backup / Standby wie unter 2 dargestellt

Vorteile:

- leicht zu erweitern

- Lastverteilung
- ausfallsicher

Nachteile:

- Webmailer muss erweitert und/oder Proxykomponente muss entwickelt werden
- Hardwareoverkill für Webmail

4.2 Mailinjection

Idee:

- nur notwendig, wenn kein Cyrus Murder verwendet wird oder wenn per SMTP eingeliefert wird
- setzt SMTP zu LMTP um (oder was auch immer der Imapserver als Protokoll erwartet)
- stellt Email anhand der Empfängeradresse auf dem richtigen Backend zu
- ggf. Bodysplitting bei Zustellung an Empfänger auf unterschiedlichen Backends

Alternative: Erweiterung von ManageMail um empfängerbasiertes Routing. Aber: Das möchten wir nicht und nicht jeder Mailboxcluster hat ein vorgeschaltetes ManageMail. Daher triviales Setup:

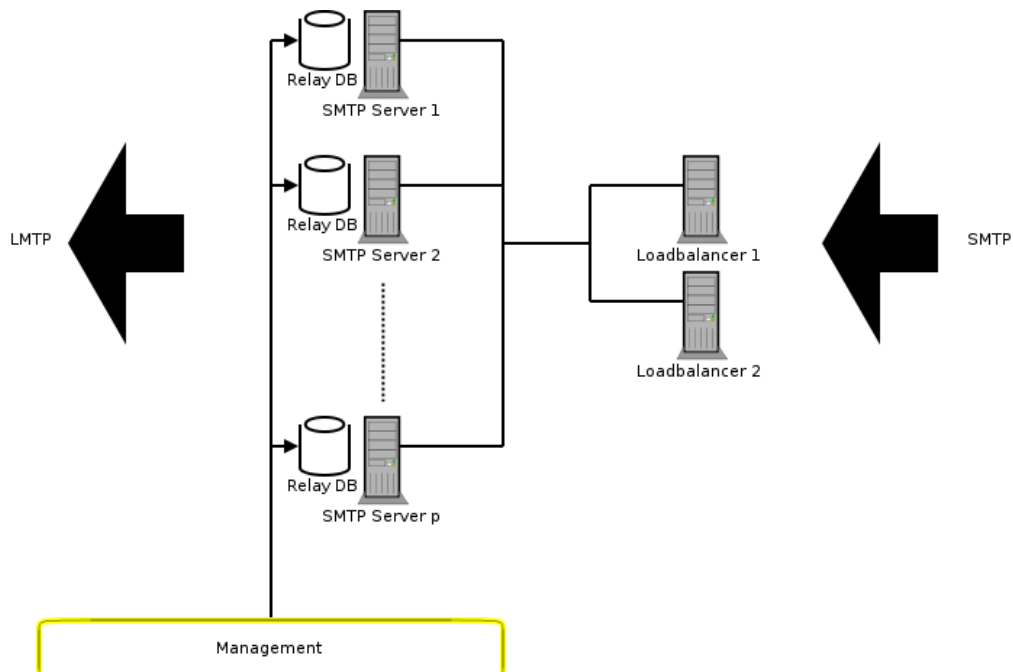


Abbildung 17: SMTP-Relay mit Loadbalancer

Die obligatorische Active-Active Cluster Alternative:

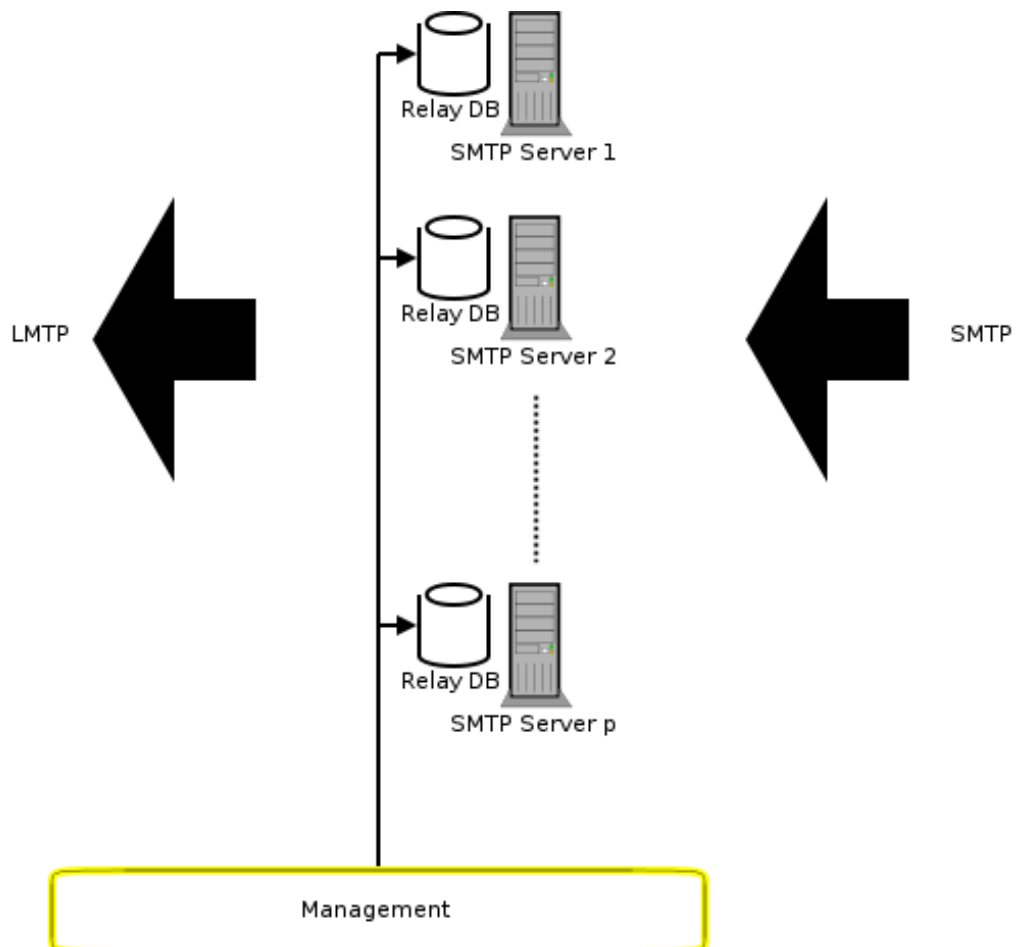


Abbildung 18: SMTP-Relay als Active-Active Cluster

5 Sinnvolles Konzept

(Ich spare mir hier separate Grafiken für Konzepte mit Loadbalancer und Active-Active Cluster, da sie betrieblich gleichwertig sind)

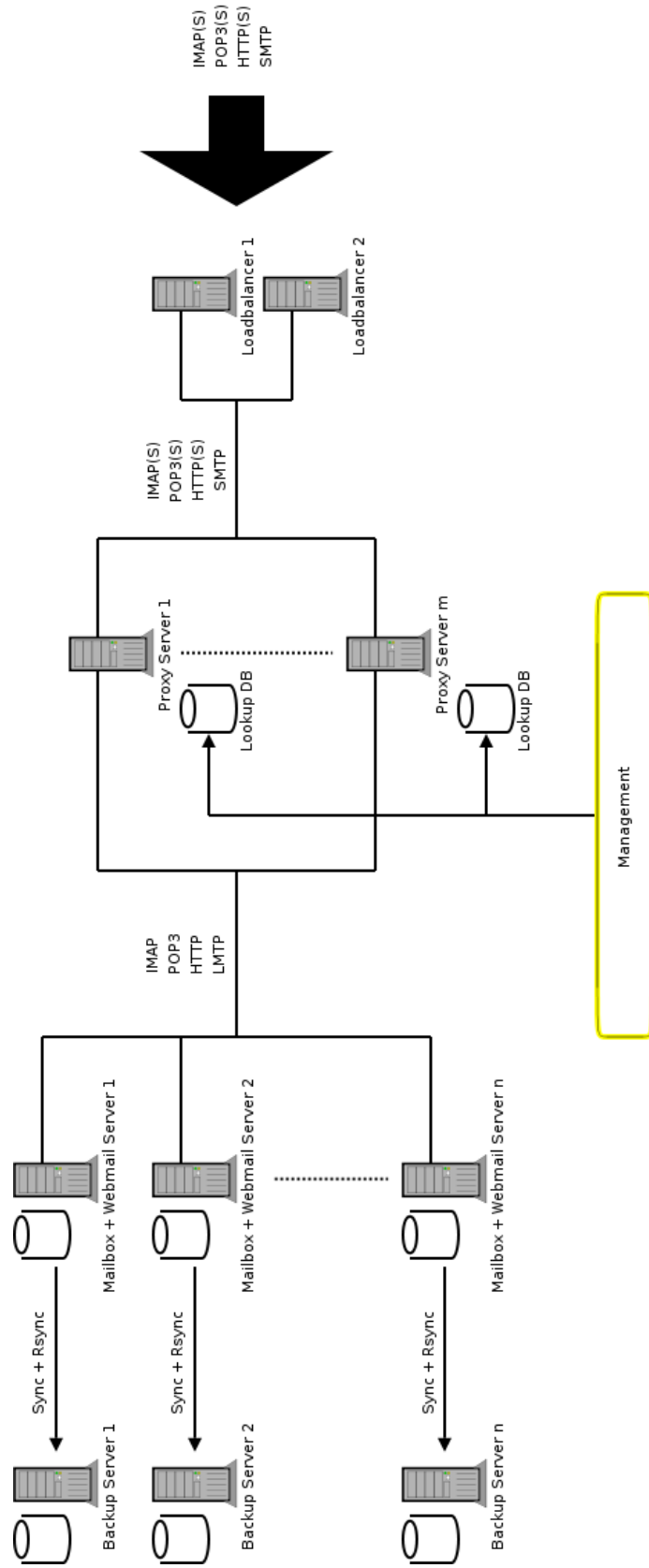


Abbildung 19: Gefaltetes Setup

Idee:

- Mailboxserver (Backend) dienen nicht nur als IMAP- und POP3-, sondern auch als Webmailserver
- sinnvollerweise liegen Postfach und Webmaileinstellungen eines Benutzers immer auf dem selben Mailboxserver
- laufendes Backup dieser Mailboxserver per Cyrus-Syncclient (Mailbox) und Rsync (Webmaileinstellungen)
- auf die Mailboxserver werden Mails ausschliesslich per LMTP eingeliefert und per IMAP und POP3 abgeholt
- dennoch muss Postfix, sendmail, o.ä. zum Versenden aus dem Webmail heraus (u.a. Vacation, Weiterleitungen) installiert sein
- Zugriff auf das Webmail erfolgt über HTTP
- vorgeschaltete Proxyserver leiten Mails bzw. Benutzer auf den richtigen Mailboxserver weiter
- Perdition terminiert IMAP(S) und POP3(S) und leitet anhand einer periodisch erzeugten Lookup-DB per IMAP bzw. POP3 weiter
- Postfix terminiert SMTP und leitet Mails an die richtigen Mailboxserver weiter
- Apache mit einer *neu zu erstellenden* Anwendung terminiert HTTP(S), übernimmt Benutzerlogin und leitet auf den richtigen Webmailserver bzw. Mailboxserver weiter
- Proxyserver wahlweise als Active-Active Cluster oder mit Loadbalancer (s. Abbildung 19)

A Quellen

Postfix <http://www.postfix.org/>

Cyrus IMAP <http://cyrusimap.web.cmu.edu/>

Perdition <http://www.vergenet.net/linux/perdition/>

Apache Webserver <http://httpd.apache.org/>

Linux Virtual Server <http://www.linuxvirtualserver.org/>

Active-Active cluster patch for FreeBSD 6.3 <http://ogris.de/carp/>