Einrichtung von s/qmail + BincIMAP

Beginner's Guide

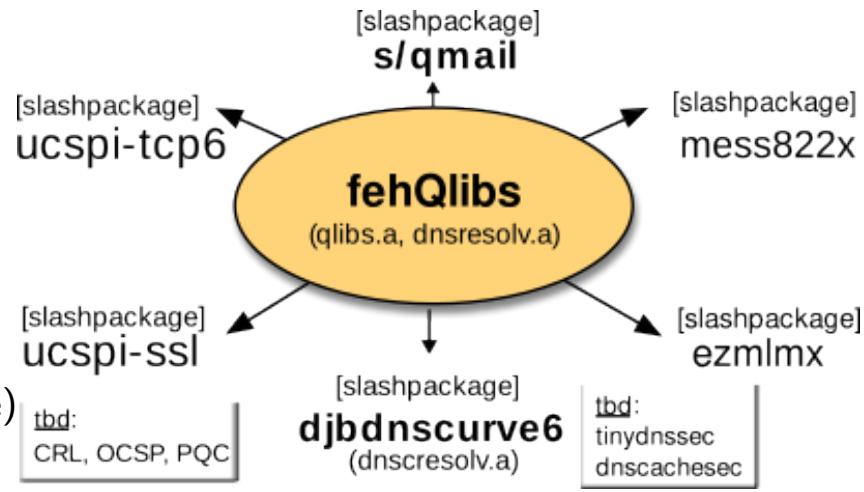
Erwin Hoffmann

www.fehcom.de

30. August 2025

Spoiler: Was man bekommt/Teil 1

- Djbware
  - fehQlibs (DJB, fefe, Kai Peter, me)
    - IP address handling IPv4/IPv6 CIDR
    - String handling `stralloc`
    - Datenstrukturen **cdb**
    - dnsresolv als Resolver-Bibliothek
    - Applikations-spezifische DNS Settings
  - ucspi-tcp6 (DJB, me)
    - Vergleichbar inetd/xinetd
    - User-space Firewall (IP und DNS)
    - Authentication via ident (obsolet)
    - rblsmtp Client mit Greetdelay
    - tcprules/tcprulescheck
  - ucspi-ssl (William Baxter, Scott Gifford, me)
    - TLS (1.3) Verbindungen
    - STARTTLS/STLS enabled
    - OpenSSL/LibreSSL
    - X.509 (Client) Certificate Handling (kein OCSP, kein CRL)
    - ucspissl Library für s/qmail
    - Self-signed X.509 Zertifikate und key files



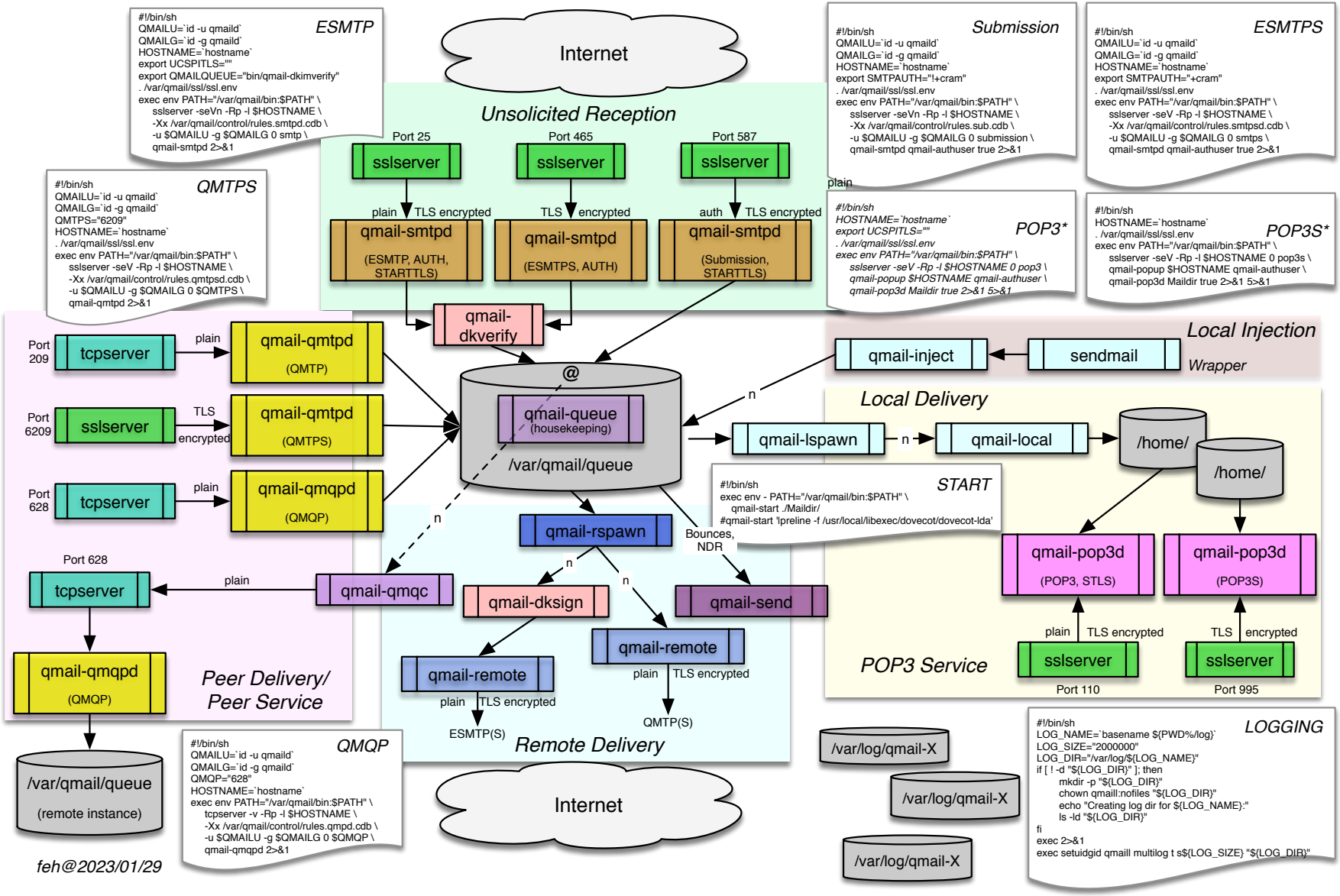
↪ 64 Bit Clean, unabhängig vom Unix OS, Big- oder Little-Endian Unterstützung

Spoiler: Was man bekommt/Teil 2

- s/qmail (DJB, fefe, viele andere, me)
  - Multi-tenant MTA auf Grundlage von DJB's Qmail
  - Durchgängige IPv6 Unterstützung incl. ULA- und LLU-Adressen
  - VERP Support (Vorläufer von BATV)
  - Gleiche APIs wie Qmail
  - Executables/Konfiguration und Queue können auf unterschiedlichen Mount-Punkten liegen (OpenBSD, OmniOS) → normalerweise `/var/qmail`
  - Recipient Validation
  - Authentifizierung und Autorisierung über X.509 Zertifikate oder IdP-Backend (via CDB, Dovecot, LDAP, Exchange ...)
  - RegEx Filter für Adressen (EHLO, Mail From, Rcpt To)
  - Filter für Mail-Inhalt/MIME Typen
  - Schnittstelle für Anti-Virus Programme (QHPSI)
  - Greylisting Client (Schweikert)
  - DKIM mit ECC (libdkim `C++`); Signieren, Verifizieren; Erzeugung von DKIM Keys
  - TLSA (ohne DANE)
  - SPF und SRS
  - (E)SMTP, Submission, SMTPS; IDN (via libidn2)
  - QMTP, QMTPS (QMTP macht sogar Postfix) → *Übung*: Welche Portnummer?
  - Basic POP3-Server
  - Ablage in Maildir oder mbox Format (User: `maildirmake`)
  - Vpopmail/Vmailmgr POP Toaster (separate Pakete)
  - MRTG Interface für Logfiles
  - DNS Client Module für Diagnose

↪ Umfangreiche Dokumentation in 13 Kapiteln zuzüglich zu den man-pages.

Spoiler: Was man bekommt/Teil 2a



Spoiler: Was man bekommt/Teil 3

- BincIMAP (Andreas Aardal Hanssen, me)
  - IMAP C++ Server auf Grundlage von ucspi-tcp6/ucspi-ssl von *Andreas Aardal Hanssen*
  - Reiner Maildir(++) Support (allerdings ohne Quota)
  - Benötigt Authentisierungs-Modul
- mess822x (DJB, me)
  - RFC822 message parser
  - Nachrichten-Header können ausgelesen werden
  - Nachrichten können in ihre MIME-Bestandteile zerlegt werden
- Ezmlmx (DJB, Bruce Guenter, Fred Lindberg, me)
  - Mailing Listen Manager auf Grundlage von VERP
  - Einfache Subscription über Email-Adressen Extension
  - Web-Frontend
  - Weitere Backends in Planung (MySQL, PostgreSQL ...)
- Newanalyse (me)
  - Shell und PERL Skript
  - Analyse der s/qmail log files (Sender/Empfänger, Viren ...)
  - Logfile Ablage und Archivierung
  - Mailflow: Wer hat wann was an wen geschickt? (Mail Transaktion)

Spoiler: Was man bekommt/Teil 4

- Djbdnscurve6 (DJB, Harm van Tilbourg, me)
  - **tinydns**: DNSCurve Verschlüsselnder UDP-basierender autoritativer DNS Server
  - **dnscache**: Entschlüsselnder UDP/TCP DNS Cache (ohne DNSSEC)
  - **rbldns**: IPv4/IPv6 Relay Black List Server (verschlüsselnd)
  - Komplette IPv6 Unterstützung mit kompaktifizierten IPv6-Adressen
  - Verschlüsselnde DNS Library

↪ SMTP nutzt umfangreich DNS-Dienste. Um Fehler zu erkennen, braucht man einen eigenen DNS Cache Server.

↪ Es kann ein eigener RBL-Dienst (*relay black list*) aufgesetzt werden, ohne permanent auf Spamhaus etc. zugreifen zu müssen.

Was man bekommt: Alles

Produkt	Datei	Version	Grösse
fehQlibs	fehQlibs-27.tgz	27	86.8K
ucspi-tcp6	ucspi-tcp6-1.13.05.tgz	1.13.05	48.8K
ucspi-ssl	ucspi-ssl-0.13.05.tgz	0.13.05	78.7K
s/qmail	sqmail-4.4.08.tgz	(alpha) 4.4.08	402K
			616.3K
Postfix	postfix-3.8.11.tar.gz	3.8.11	4.4M
mess822x	mess822x-1.24.tgz	1.24	65.4K
ezmlmx	ezmlmx-0.68a.tgz	(unpublished) 0.68	593K
djbdnscurve6	djbdnscurve6-45.tgz	(3.)45	123K
BincIMAP	bincimap-2.0.16.tgz	2.0.16	145K
Dovecot	dovecot-2.4.1-4.tar.gz	2.4.2	6.0M

Tabelle: Name, Version und Grössen der gepackten Quelldateien

Installation

- Generell: slashpackage Format
- `mkdir -p /package; chmod 1755 /package`
- `cd package`
- `tar -xzf path/ucspi-tcp6-1.13.05.tgz` → `net/ucspi-tcp6`
- `cd net/ucspi-tcp6; ls; cd ucspi-tcp6-1.13.05`
- `package/install`
- Andere Präfixe:
  - `/package/mail/sqmail-xyz`
  - `/package/net/djbdnscurve6-xzy`
  - `/package/host/superscript.com/net/ucspi-ssl`
- Damit automatische Versionierung mit einfachem Fall-Back; Semantic Versioning
- Blick in README.md, INSTALL.md und CHANGELOG werfen!
- Compiliert wird von Source → [gcc, clang] + make reicht; `package/compile`
- Kein `./configure` Schritt
- Ggf. werden notwendige Modifikationen über `./conf-XX` Dateien vorgenommen
- Ablage der Executables konfigurierbar; typischerweise `/usr/local/bin`
- Für viele Pakete reicht ein `package/install`

↪ Ausnahme: *fehQlibs*; reine Library (+ man pages) mit statischem oder dynamischen Binden. Liegen typischerweise unter `/usr/local/qlibs -> /usr/local/fehQlibs-XX`

Ersteinrichtung

- User Einrichtung: **s/qmail** + **djbdnscurve6** brauchen dedizierte Benutzer
- `/package/ids` (oder so) zu installieren
- `./conf-XX` Dateien liegen vor, um Namen und UID/GID und Pfade anzupassen
- Compiler und Linker Optionen können per `conf-cc` und `conf-ld` modifiziert werden
- Ablage der man pages kann customized werden falls `$manpath` nicht verfügbar (`conf-man`); man oder Mandoc Format; `package/man`
- ucspi-ssl kann mit unterschiedlichen OpenSSL/LibreSSL Versionen umgehen; auch Versionen, die nicht OS-Standard sind (`conf-ssl`, `conf-ssllib`, `conf-ucspissl`)
- Fertige, rudimentäre run-Skripte für Daemontools mit Einrichtung der Services; `package/svc`

Betrieb

- Im laufenden Betrieb müssen *Daemons* aufgesetzt werden:
  - qmail-send – Mails versenden
  - qmail-smtpd – Mails empfangen (StartTLS, Port 25)
  - qmail-smtpsd – Mails empfangen (TLS, Port 465 oder 587)
  - qmail-pop3sd – POP3 Zugriff (TLS, Port pops) → *Übung*: Welche Nummer?
  - bincimapd – IMAP4 Zugriff (TLS, Port imaps) → *Übung*: Welche Nummer?
- Hierfür werden ./run Skripte mitgeliefert, die unter
  - *DJB*'s Daemontools, oder
  - *Gerrit Pape*'s **runit**, oder auch unter
  - *Poettering*'s **systemd**

  eingesetzt werden können.
- s/qmail kommt mit **systemd** `unit`-Files.
- Aufgaben:
  - Start der Daemons
  - Überwachen
  - Herunterfahren (und neu Konfigurieren)
  - Automatischer Neustart bei Fehlfunktion

Konfiguration/Teil 1

- ucspi-tcp6/ucspi-ssl:
  - Erstellung der IP-Verbindungstabellen (`allow`/`deny`) für tcpserver/sslserver (rückwärtskompatibel mit DJB's ucspi-tcp)
  - Einträge pro IP, IP-Netz (CIDR) oder Domain (FQDN)
  - Flagging als `allow` oder `deny`
  - Bei `allow`: Setzen von Environment-Parameter für aufrufendes Programm
  - Werkzeuge: **tcprules**, **tcprulescheck**
- tinydns:
  - Aufbau der Zonendatei (splitted view) und Compilieren
  - Neue Einträge können per CLI eingerichtet und aktiviert werden (persistiert)
  - Erzeugen von ECC Private- und Public Key für Verschlüsselung
  - Binden auf IP und Port (Öffentlich oder splitted view)
- dnscache:
  - Binden auf IP und Port
  - Zulassen der abfragenden IPs (dnscache ist nicht Öffentlich)
  - Eintragen der Root-Server + eigener Zonen + Nameserver
  - dnscache arbeitet ausschliesslich im RAM ($\rightarrow$ Grösse beschränken)
- BincIMAP:
  - Keine weitere Konfiguration erforderlich (geht über ucspi-ssl)

Konfiguration/Teil 2

- s/qmail:
  - `/var/qmail/control`: Run-time Konfiguration; `qmail-showctl`
  - `/var/qmail/users`: Erlaubte Benutzer (`authuser`) und Benutzer-Rewriting (Email address → Unix User)
  - `/var/qmail/ssl`: TLS Zertifikate und Private Keys (Links to Lets-Encrypt certs)
  - `/var/qmail/ssl/domainkeys`: DKIM Material pro Domäne
  - `/var/qmail/alias`: Alias-User Funktionalität (`.qmail-root`, `.qmail-postmaster`)

```
/var/qmail$ ls -la
total 48
drwxr-xr-x 12 root   sqmail  4096 Aug 29  2023 .
drwxr-xr-x 17 root   root    4096 Jul 10  2024 ..
drwxr-sr-x  2 alias  sqmail  4096 Jun  3 13:18 alias
drwxr-xr-x  2 root   sqmail  4096 Jul  8 10:52 bin
drwxr-xr-x  2 root   sqmail  4096 May 20 13:55 control
drwxr-xr-x  2 root   root    4096 Sep  1  2022 doc
drwxr-xr-x  3 root   root    4096 Aug 27 08:28 etc
drwxr-x--- 12 qmailq sqmail  4096 Dec 26  2022 queue
drwxr-xr-x  6 sqmtls nofiles 4096 Aug 12 20:50 ssl
drwxr-xr-x 15 root   root    4096 Sep  2  2022 svc
drwxr-xr-x  2 root   sqmail  4096 Jun  1 10:04 users
```

Wartung

Jeder im Internet öffentlich sichtbarer und ansprechbarer Daemon muss überwacht werden!

- Wir unterscheiden:
  - Logging – Schreiben der Logs für Post-Analyse
  - Monitoring – Feststellen von Engpässen
  - Alarming – Erkennung von und Benachrichtigung bei Angriffen
  - Archivierung – Ablage der Logfiles
- Alle meine Tools nehmen ein Logging auf `STDOUT`/`STDERR` (und manchmal `FD5`) vor
  - Einbindung über Logdaemons (syslog, newsyslog)
  - Logging über **Systemd** ist kontra-produktiv: Mehrere Gigabyte/Tag!
  - Bevorzugt ist DJB's **multilog**; automatische Logfile Rotation, tai64 Zeitstempel ($\rightarrow$ Newanalyse)
- Queue Management:
  - `qmail-qread`, `qmail-qstat`, `qmail-tcpto`
  - `qmail-qmaint`
- DNS Management:
  - `dnscname`, `dnsip`, `dnsptr`, `dnstxt`
    $\rightarrow$ *Übung*: `mail._domainkey.start.plasticbakjes.shop`
  - `dnsfq`, `dnsmxip`, `dnstlsa`
    $\rightarrow$ *Übung*: MX + TLSA Record für die obige Domain

$\hookrightarrow$ Wird s/qmail über Systemd eingesetzt, kann über eine geeignete cgroup Performance-Parameter, z.B. über Prometheus erfolgen

Theorie → Praxis

Let's do it!