Piler

- Installation von Piler Mailarchive auf Debian Stretch

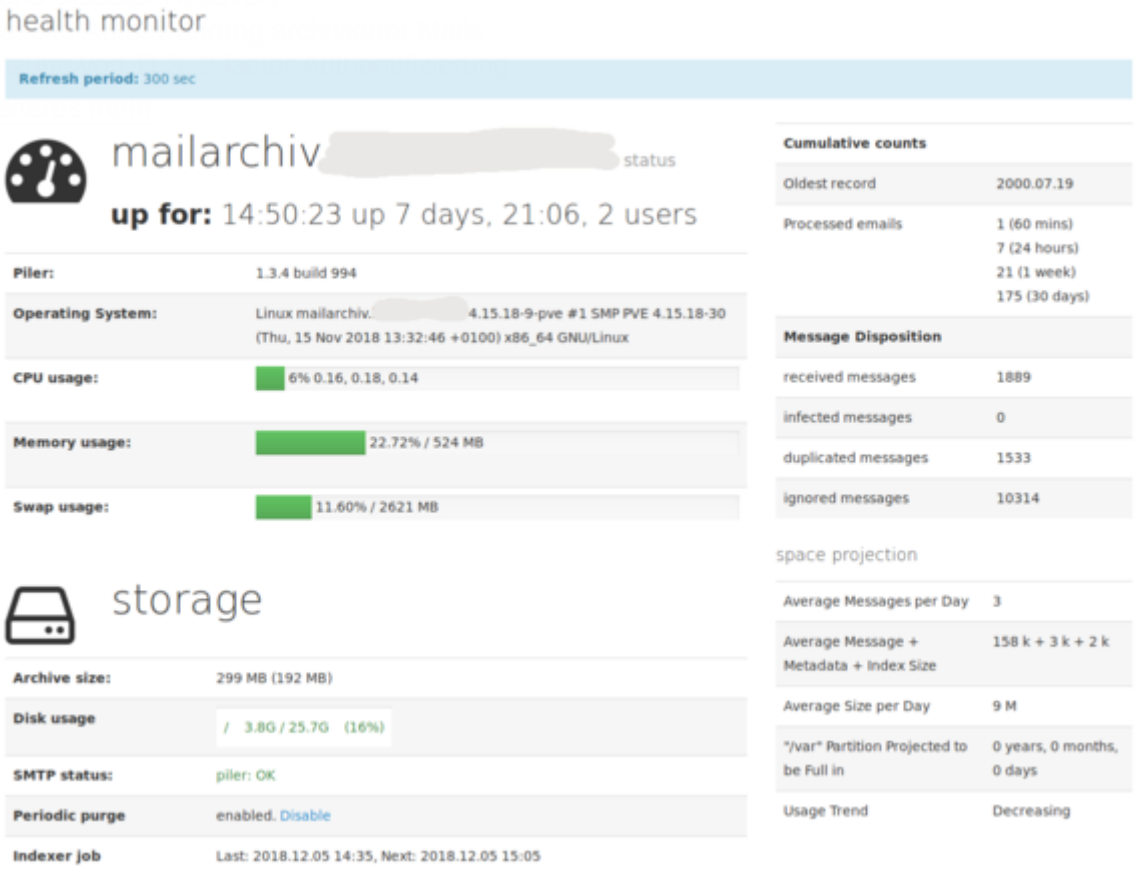
Installation von Piler Mailarchive auf Debian Stretch

Jeder Gewerbetreibende muss Geschäftskorrespondenz für mind. 6 bzw. bis zu 7 Jahre aufheben (Quelle 1, Quelle 2). Wer jetzt nicht all seine Emails ausdrucken und abheften möchte benötigt ein Mailarchiv. Dazu werden einfach alle ein- und ausgehenden Emails zusätzliche zum jeweiligen Empfänger auch an das Archiv geschickt. Ich nutze dafür die Software `piler`, die auf einer virtuellen Maschine läuft. Nach der Installation lauscht piler auf Port 25, nimmt für die konfigurierten Domains Emails an und archiviert diese fälschungssicher.

Piler ist Open Source, bietet eine PHP basierte GUI und unterstützt:

- verschiedene Archivierungs- und Aufbewahrungsregeln
- Deduplikation
- digitaler Fingerprint und Verifizierung
- Volltextsuche in archivierten Mails
- Trennung von Administratoren- und Auditoren- Accounts
- Unterstützung von Google Apps und Office 365
- (Erst-)Import von Mails via IMAP
- Backup und Disaster Recovery
- verschl
- Unterst
- vieles w

Screenshot:



Voraussetzungen

- Ausreichend Diskspace für das Mailarchiv
- Frisches Debian Stretch System, 64-bit
- MariaDB 10.1+
- Sphinx Search 2.2.x
- PHP 7.x
- nginx (geht auch mit Apache oder anderen Webservern)
- ZIP
- mpstat
- python mit mysql

Als Hardware habe ich folgendes gewählt, kommt bei euch natürlich auch etwas auf die Anzahl der Mails an, die verarbeitet werden sollen.

- 2 vCPUs
- 4 GB RAM
- 80 GB Disk

Vor der Installation

Zuerst sollte ein DNS-Eintrag für Piler angelegt werden. Dieser wird benötigt um Mails an Piler weiterleiten zu können und für die GUI.

Installation

Debian Backports aktivieren:

```
deb http://ftp.debian.org/debian stretch-backports main
```

`apt update && apt upgrade`

Postfix muss entfernt werden, da Piler selbst auf Port 25 lauschen mag: `apt purge postfix`

Jetzt müssen noch die notwendigen Pakete installiert werden:

```
apt install nginx openssl libssl-dev mariadb-server default-libmysqlclient-dev sphinxsearch memc
 build-essential python-mysqldb php7.0-fpm php7.0-curl php7.0-gd php7.0-mysql php7.0-cli php7.0-
 php7.0-ldap php7.0-mbstring php-memcached libtre-dev sysstat gcc libwrap0 libwrap0-dev latex2rt
 catdoc poppler-utils unrtf tnef unixodbc libpq5 libzip-dev libzip4 zipcmp zipmerge ziptool
```

Der automatische Start von Sphinx muss ausgeschaltet bleiben, bitte prüfe ob in `/etc/default/sphinxsearch` der Eintrag `„START=no“` hinterlegt ist.

Gruppe und Benutzer anlegen:

```
groupadd -r piler
useradd -r -g piler -m -s /bin/sh -d /var/piler piler
usermod -L piler
chmod 0755 /var/piler
```

Piler herunterladen und kompilieren. Den aktuellen Tarball gibts immer hier. Als root:

```
wget https://bitbucket.org/jsuto/piler/downloads/piler-1.3.4.tar.gz
tar xvzf piler-1.3.4.tar.gz
cd piler-1.3.4/
./configure \
```

```
  --localstatedir=/var \
  --with-database=mysql \
  --enable-tcpwrappers \
  --enable-memcached
make
make install
ldconfig
```

Konfiguration

Damit der Postinstall problemlos durchläuft sollte die root-Shell noch von dash auf bash gesetzt werden: `dpkg-reconfigure dash` –> NO auswählen (also bash verwenden). Danach einmal aus- und wieder einloggen.

Der Postinstall kommt leider noch nicht mit dem MySQL auth-socket ab Debian Stretch klar, daher habe ich hier eine gepatchte Version zur Verfügung gestellt. Der Postinstall kümmert sich um die Einrichtung der MySQL-Datenbank, Sphinx, ggf. einem Smarthost/Relayhost und die WebGUI.

```
make postinstall


This is the postinstall utility for piler
It should be run only at the first install. DO NOT run on an existing piler installation!


Continue? [Y/N] [N] y


Please enter the webserver groupname [www-data]

Please enter mysql hostname [localhost]

Please enter mysql socket path [/var/run/mysqld/mysqld.sock]

Please enter mysql database [piler]

Please enter mysql user name [piler]

Please enter mysql password for piler [] <geheimespasswort>
mysql connection successful


Please enter the path of sphinx.conf [/usr/local/etc/piler/sphinx.conf]

Please enter smtp relay []

Please enter smtp relay port [25]
no crontab for piler



INSTALLATION SUMMARY:

piler user: piler
keyfile: /usr/local/etc/piler/piler.key

mysql host: localhost
mysql socket: /var/run/mysqld/mysqld.sock
mysql database: piler
mysql username: piler
mysql password: *******

sphinx indexer: /usr/bin/indexer
sphinx config file: /usr/local/etc/piler/sphinx.conf

vhost docroot: /var/www/piler
www group: www-data

smtp relay host:
smtp relay port: 25

piler crontab:
### PILERSTART
5,35 * * * * /usr/local/libexec/piler/indexer.delta.sh
30   2 * * * /usr/local/libexec/piler/indexer.main.sh
```

```
15,45 * * * * /usr/local/libexec/piler/indexer.attachment.sh
*/15 * * * * /usr/bin/indexer --quiet tag1 --rotate --config /usr/local/etc/piler/sphinx.conf
*/15 * * * * /usr/bin/indexer --quiet note1 --rotate --config /usr/local/etc/piler/sphinx.conf
30   6 * * * /usr/bin/php /usr/local/libexec/piler/generate_stats.php --webui /var/www/piler >/d
*/5 * * * * /usr/bin/find /var/www/piler/tmp -type f -name i.\* -exec rm -f {} \;
### PILEREND



Correct? [Y/N] [N] y

Continue and modify system? [Y/N] [N] y

Creating mysql database... Done.
Writing sphinx configuration... Done.
Initializing sphinx indices... Sphinx 2.2.11-id64-release (95ae9a6)
Copyright (c) 2001-2016, Andrew Aksyonoff
Copyright (c) 2008-2016, Sphinx Technologies Inc (http://sphinxsearch.com)

using config file '/usr/local/etc/piler/sphinx.conf'...
indexing index 'main1'...
collected 0 docs, 0.0 MB
total 0 docs, 0 bytes
total 0.002 sec, 0 bytes/sec, 0.00 docs/sec
indexing index 'main2'...
collected 0 docs, 0.0 MB
total 0 docs, 0 bytes
total 0.000 sec, 0 bytes/sec, 0.00 docs/sec
indexing index 'main3'...
collected 0 docs, 0.0 MB
total 0 docs, 0 bytes
total 0.000 sec, 0 bytes/sec, 0.00 docs/sec
indexing index 'main4'...
collected 0 docs, 0.0 MB
total 0 docs, 0 bytes
total 0.000 sec, 0 bytes/sec, 0.00 docs/sec
indexing index 'dailydelta1'...
collected 0 docs, 0.0 MB
total 0 docs, 0 bytes
total 0.000 sec, 0 bytes/sec, 0.00 docs/sec
indexing index 'delta1'...
collected 0 docs, 0.0 MB
total 0 docs, 0 bytes
total 0.002 sec, 0 bytes/sec, 0.00 docs/sec
indexing index 'tag1'...
collected 0 docs, 0.0 MB
total 0 docs, 0 bytes
total 0.000 sec, 0 bytes/sec, 0.00 docs/sec
indexing index 'note1'...
collected 0 docs, 0.0 MB
total 0 docs, 0 bytes
total 0.000 sec, 0 bytes/sec, 0.00 docs/sec
indexing index 'att1'...
collected 0 docs, 0.0 MB
total 0 docs, 0 bytes
total 0.001 sec, 0 bytes/sec, 0.00 docs/sec
total 9 reads, 0.000 sec, 0.0 kb/call avg, 0.0 msec/call avg
total 63 writes, 0.000 sec, 0.0 kb/call avg, 0.0 msec/call avg
Done.
installing cron entries for piler... Done.
installing keyfile (piler.key) to /usr/local/etc/piler/piler.key... Done.
Making an ssl certificate ... Generating a RSA private key
.....................................................................................
.....................................................................................
writing new private key to '/usr/local/etc/piler/piler.pem'
-----
Copying www files to /var/www/piler... Done.

Done post installation tasks.
```

In der Konfigurationsdatei `/usr/local/etc/piler/piler.conf` muss noch die passende hostid hinterlegt werden. Dazu wird die Datei im Texteditor geöffnet und etwa in Zeile 14 die hostid auf den Hostnamen deines Pilers gesetzt (z.B. piler.mydomain.de). Außerdem legen wir hier gleich den Zeitraum fest für den Mails aufgehoben werden sollen: `default_retention_days=2557`.

Außerdem scheint beim postinstall-Script eine Variablenersetzung nicht zu klappen, daher muss noch die Konfiguration der WebUI angepasst werden. In `/var/www/piler/config.php` etwa in Zeile 321 steht `${prefix}`

das durch den vollständigen Pfad zur config-site.php ersetzt werden muss, z.B:

```
-require_once '${prefix}/etc/piler/config-site.php';
+require_once '/var/www/piler/config-site.php';
```

Der nginx-Vhost schaut bei mir so aus:

```
server {
  server_name piler.mydomain.de;

  root /var/www/piler;

  access_log /var/log/nginx/access.log;
  error_log /var/log/nginx/error.log;

  gzip on;
  gzip_types text/plain application/xml text/css;
  gzip_vary on;

  location / {
    index  index.php index.html;
    try_files $uri $uri/ /index.php;
  }

  error_page   500 502 503 504  /50x.html;
  location = /50x.html {
    root   html;
  }

  location ~ [^/]\.php(/|$) {
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;

    fastcgi_split_path_info ^(.+?\.php)(/.*)$;
    if (!-f $document_root$fastcgi_script_name) {
      return 404;
    }

    fastcgi_pass unix:/var/run/php/php7.0-fpm.sock;
    fastcgi_index index.php;
    include fastcgi_params;
  }

  location ~* \.(ico|css|js|gif|jpe?g|png)$ {
    expires 2w;
  }

  rewrite /search.php /index.php?route=search/search&type=simple;
  rewrite /advanced.php /index.php?route=search/search&type=advanced;
  rewrite /expert.php /index.php?route=search/search&type=expert;
  rewrite /search-helper.php /index.php?route=search/helper;
  rewrite /audit-helper.php /index.php?route=audit/helper;
  rewrite /message.php /index.php?route=message/view;
  rewrite /bulkrestore.php /index.php?route=message/bulkrestore;
  rewrite /bulkremove.php /index.php?route=message/bulkremove;
  rewrite /bulkpdf.php /index.php?route=message/bulkpdf;
  rewrite /folders.php /index.php?route=folder/list&;
  rewrite /settings.php /index.php?route=user/settings;
  rewrite /login.php /index.php?route=login/login;
  rewrite /logout.php /index.php?route=login/logout;
  rewrite /google.php /index.php?route=login/google;
  rewrite /domain.php /index.php?route=domain/domain;
  rewrite /ldap.php /index.php?route=ldap/list;
  rewrite /customer.php /index.php?route=customer/list;
  rewrite /retention.php /index.php?route=policy/retention;
  rewrite /archiving.php /index.php?route=policy/archiving;
  rewrite /legalhold.php /index.php?route=policy/legalhold;
  rewrite /view/javascript/piler.js /js.php;
}
```

Jetzt nicht vergessen den Vhost zu aktivieren und nginx zu reloaden:

```
ln -s /etc/nginx/sites-available/test-piler1.wavecloud.de /etc/nginx/sites-enabled/
nginx -t
nginx -s reload
```

Init-Scripte (Systemd-Unit files werden keine mitgeliefert) kopieren und aktiveren:

```
cp init.d/rc.piler /etc/init.d/
cp init.d/rc.searchd /etc/init.d
systemctl daemon-reload
systemctl enable rc.piler.service
systemctl enable rc.searchd.service
systemctl start rc.piler.service
systemctl start rc.searchd.service
```

Die GUI sollte nun über den Browser erreichbar sein: `http://piler.mydomain.de/`. Es empfiehlt sich der Seite noch eine passende SSL/TLS-Konfiguration zu spendieren.

Die Standardbenutzername lautet **admin@local** / **pilerrocks** und sollte gleich geändert werden. Es sollte gleich auch ein Kennwort für den Standardauditor gesetzt werden. Die Rolle „Master admin“ darf verwalten und administrieren. Die Rolle „Auditor“ wird verwendet um auf Mails im Archiv zuzugreifen. Beide haben unterschiedliche Oberflächen.

Ich habe noch folgende Einstellungen in der GUI vorgenommen:

- administration ? domain: Hinzugefügt `ovtec.it` / mapped domain: `ovtec.it`
- administration ? archiving rules: Hier werden Regeln eingetragen, deren Ziele NICHT archiviert werden (z.B. CRON-Mails usw.)
- administration ? retention rules: Ist hier nicht eingetragen gilt der Standardwert aus der piler.conf (2557 Tage)

Der Host sollte nun noch einmal abschließend rebootet werden, danach prüfen ob alle Dienste laufen:

```
# ps ax | grep -E "piler|searchd|mysql|nginx"
  445 ?        Ssl    0:00 /usr/sbin/mysqld
  448 ?        S      0:00 searchd --config /usr/local/etc/piler/sphinx.conf
  449 ?        Sl     0:00 searchd --config /usr/local/etc/piler/sphinx.conf
  453 ?        Ss     0:00 nginx: master process /usr/sbin/nginx -g daemon on; master_process on
  454 ?        S      0:00 nginx: worker process
  455 ?        S      0:00 nginx: worker process
  505 ?        Ss     0:00 /usr/local/sbin/piler-smtp -d
  508 ?        Ss     0:00 /usr/local/sbin/piler -d
  509 ?        S      0:00 /usr/local/sbin/piler -d
  510 ?        S      0:00 /usr/local/sbin/piler -d
```

Alle Mails, die nun an bspw. archive@piler.mydomain.de geschickt werden und mit einer der in `Domains` konfigurierten Domains übereinstimmen werden entsprechend den eingestellten Regeln archiviert. Es gibt verschiedene Wege, wie Mails weitergeleitet werden können. Die Piler-Dokumentation führt dazu einige auf.

Dokumentation

Hier noch ein paar Links zur Piler Dokumentation:

- Installation
- FAQ
- Übersicht