

MailPiler-Installation

Schritt-für-Schritt-Anleitung

Mail Piler ist eine Open-Source Software zur Archivierung von E-Mails. Im Business Umfeld muss zum Beispiel jede E-Mail für 10 Jahre archiviert werden. In der folgenden Anleitung werde ich die Installation unter Rocky Linux 9 vornehmen. Da Rocky von RedHat abstammt, bietet es einen sehr langen Support und eignet sich hervorragend für Archivierungsanwendungen. Die Distro bietet noch offiziellen Support bis 2032.

Installation des OS

Als erstes wird eine Installations ISO benötigt. Ich habe meiner VM 2 Kerne, 2GB RAM und 32GB Speicher zugewiesen, diese Ressourcen können aber auch jederzeit angepasst werden. Unter Proxmox (auf einem anderen Hypervisor habe ich es noch nicht getestet) funktioniert die Installation nur, wenn der CPU Typ auf Host gestellt ist. Nachdem der Installer gebootet ist, müssen folgende Einstellungen festgelegt werden:

- Sprache Tastaturlayout (en entfernen, DE hinzufügen)
- root Passwort
- Software Selection => Minimal Install
- Zielfestplatte

Nun kann das Betriebssystem installiert werden. Nach dem ersten Starten muss das System noch auf den aktuellsten Stand gebracht werden.

Dies geschieht mit dem folgenden Befehl:

```
dnf update
```

Nun müssen die extra Packages für RHEL aktiviert werden.

Dies geschieht mit dem folgenden Befehl:

```
dnf -y install epel-release
```

Nachdem wir die extra Pakete installiert haben, geht es mit ein paar allgemeinen Tools, die auf jedem System meiner Meinung nach installiert werden sollten, weiter:

```
dnf install htop wget nano git tar
```

Nachdem dies erledigt ist, müssen wir der VM noch einen Hostnamen geben. Dazu die Datei `/etc/hostname` bearbeiten und den Namen eintragen (bspw. mit nano). Abschließend muss zum Übernehmen des Namens noch die VM neugestartet werden.

Abhängigkeiten installieren

Nachdem das Betriebssystem nun aufgesetzt ist, kann mit der eigentlichen Installation begonnen werden. Ein Teil der Abhängigkeiten kann direkt als Paket heruntergeladen werden, der Rest muss so auf dem System installiert werden.

```
dnf install https://repo.manticoresearch.com/manticore-repo.noarch.rpm
dnf install openssl mariadb mariadb-server php php-ldap php-gd php-memcache php-pdo php-mysqli
php-curl php-zip nginx manticore manticore-extra tre-devel sysstat python3 gcc poppler libytnef-devel memcached
tcp_wrappers-libs openssl-devel zlib-devel mariadb-connector-c-devel
systemctl enable --now sysstat
wget http://ftp.wagner.pp.ru/pub/catdoc/catdoc-0.95.tar.gz tar
-xzf catdoc-0.95.tar.gz
cd catdoc-0.95
./configure
make
make install
cd ..
rm catdoc-0.95 -Rf
dnf install http://repo.iotti.biz/CentOS/9/x86_64/unrtf-0.21.9-17.el9.lux.x86_64.rpm
http://repo.iotti.biz/CentOS/7/x86_64/xlhtml-0.5-19.el7.lux.x86_64.rpm
https://yum.oracle.com/repo/OracleLinux/OL7/latest/x86_64/getPackage/tcp_wrappers-devel-7.6-
77.el7.x86_64.rpm
https://yum.oracle.com/repo/OracleLinux/OL7/latest/x86_64/getPackage/tcp_wrappers-libs-7.6-
77.el7.x86_64.rpm https://dl.rockylinux.org/pub/rocky/9/CRB/x86_64/os/Packages/l/libzip-devel-
1.7.3-7.el9.x86_64.rpm
systemctl start mariadb
systemctl enable mariadb
systemctl start nginx
systemctl enable nginx
systemctl disable manticore
systemctl stop manticore
chown nginx:nginx /var/lib/php/session -R
```

SELinux deaktivieren

```
nano /etc/sysconfig/selinux
SELINUX=disabled
/etc/selinux/config
```

SELINUX=disabled

Firewall Ports öffnen

```
firewall-cmd --zone=public --permanent --add-port=80/tcp
```

```
firewall-cmd --zone=public --permanent --add-port=25/tcp
```

Nun muss noch in `/etc/php-fpm.d/www.conf` der `user` und der `group` Parameter auf `nginx` geändert werden. Anschließend den PHP-FPM Prozess mit `systemctl restart php-fpm` neu starten.

Piler installieren

Nun sollten alle Abhängigkeiten auf dem System installiert sein und wir können mit der Installation des Pilers an sich weitermachen.

Dazu muss als Erstes ein neuer Nutzer angelegt werden:

```
groupadd piler
```

```
useradd -g piler -m -s /bin/bash -d /var/piler piler
```

```
usermod -L piler
```

```
chmod 755 /var/piler
```

Nachdem der Nutzer angelegt ist, können wir weiterhin als root Piler bauen und installieren:

```
wget https://bitbucket.org/jsuto/piler/downloads/piler-1.4.1.tar.gz
```

```
tar -xzf piler-1.4.1.tar.gz
```

```
cd piler-1.4.1
```

```
./configure --localstatedir=/var --with-database=mariadb
```

```
make
```

```
make install
```

```
echo "/usr/local/lib" >> /etc/ld.so.conf
```

```
ldconfig
```

Nun ist Mail Piler auf dem System installiert. Zur initialen Konfiguration müssen wir nun mit `OPENSSL_ENABLE_SHA1_SIGNATURES=1 make postinstall` den Konfigurationsassistenten durchlaufen. Dabei müssen die folgenden Optionen eingegeben werden:

WebServer Gruppe: nginx

MySQL Hostname: localhost

MySQL Socket: /var/lib/mysql/mysql.sock

MySQL Database: bestätigen

MySQL Nutzernamen: bestätigen

MySQL Piler Passwort: Zufallspasswort generieren

MySQL Root Passwort: egal was, da socket Authentifizierung verwendet wird

SMTP Relay Host: öffentlicher Domain des Mail Piler

SMTP Port: 25

2x mit y bestätigen

Das Script sollte nun erfolgreich durchlaufen und zum Abschluss die Meldung `Done post installation tasks.` bringen. Nachdem der eigentliche Piler jetzt läuft, müssen wir den WebServer konfigurieren. Dazu habe ich eine fertige Konfiguration, welche wir nur einspielen müssen und anschließend den Nginx neustarten:

```
rm /etc/nginx/nginx.conf
```

```
wget -O /etc/nginx/nginx.conf https://jonasled.de/download/mailpiler-nginx.conf
```

```
systemctl restart nginx
```

Wenn alles gut gelaufen ist, sollte nun das Login über den Web Browser erreichbar sein. Auf der Oberfläche können wir uns nun mit `admin@local` als Nutzernamen und `pillerocks` als Passwort anmelden. Falls es nach dem Login zu einem Redirect auf eine falsche Domain kommt, muss in der Datei `/usr/local/etc/piler/config-site.php` der `SITE_NAME` geändert werden. Hier wird aber noch nicht viel funktionieren, da die Dienste noch nicht funktionieren. Dazu müssen zuerst die Services und einige Dateien an der richtigen Stelle verlinkt werden und danach die Services gestartet werden:

```
ln -s /usr/local/sbin/piler /usr/sbin/piler
```

```
ln -s /usr/local/sbin/piler-smtp /usr/sbin/piler-smtp
```

```
ln -s /usr/local/etc/piler /etc/piler
```

```
ln -s /usr/local/etc/piler/manticore.conf /usr/local/etc/piler/sphinx.conf
```

```
ln -s /usr/local/libexec/piler/pilersearch.service /etc/systemd/system/pilersearch.service
```

```
ln -s /usr/local/libexec/piler/piler.service /etc/systemd/system/piler.service
```

```
ln -s /usr/local/libexec/piler/piler-smtp.service /etc/systemd/system/piler-smtp.service
```

```
systemctl daemon-reload
```

```
systemctl enable --now piler
```

```
systemctl enable --now pilersearch
```

```
systemctl enable --now piler-smtp
```

Nachdem die Dienste laufen, sollte nun auch die Administrator Oberfläche funktionieren. Hier kann auch unter `administration => users` ein Passwort für den auditor Nutzer festgelegt werden. Dieser darf alle E-Mails sehen. Um den Piler zu testen, können wir nun testweise eine EMail an diesen schicken. Die Empfängeradresse ist hierbei egal.

Da der Piler nicht dauerhaft die E-Mails aktualisiert, kann dies nun bis zu einer Stunde dauern, bis die Test E-Mail in der Oberfläche auftaucht.

Um dies zu beschleunigen, kann man auch die entsprechenden Befehle als Piler Nutzer ausführen:

```
# Als Piler Nutzer anmelden
```

```
su piler
```

```
/usr/bin/indexer --quiet tag1 --rotate --config /usr/local/etc/piler/manticore.conf
```

```
/usr/bin/indexer --quiet note1 --rotate --config /usr/local/etc/piler/manticore.conf
```

```
/usr/local/libexec/piler/indexer.delta.sh
```

Danach sollte in der Oberfläche, wenn man als Auditor Nutzer angemeldet ist, eine E-Mail erscheinen.

Login via IMAP

Damit sich auch Besitzer einer Mailbox am Piler anmelden können und auch E-Mails aus dem Piler wiederherstellen können, kann sich der Piler gegen einen IMAP Server authentifizieren. Dazu muss in der Datei `/usr/local/etc/piler/config-site.php` folgender Inhalt hinzugefügt und entsprechend angepasst werden:

```
$config['ENABLE_IMAP_AUTH'] = 1;
```

```
$config['RESTORE_OVER_IMAP'] = 1;
```

```
$config['IMAP_RESTORE_FOLDER_INBOX'] = 'INBOX';
```

```
$config['IMAP_RESTORE_FOLDER_SENT'] = 'Sent';
```

```
Login via IMAP$config['IMAP_HOST'] = '<HOST DES E-MAIL SERVERS>';
```

Revision #2

Created 27 October 2024 02:20:00 by Christian

Updated 5 December 2025 07:40:04 by Christian