

# Server Dokumentation

## Secure CentOS 8.3 Server aufsetzen

Todo:

Checklist bauen

Script um vieles automatisch zu setzen

Magento Stuff

Von: Dipl.Wirt.-Ing. Nick Herrmann

(Stand: Juli 2021)

# Inhalt

## Grundeinstellungen

- Centos 8.3 Minimal Einleitung S. 3
- EPEL Repository S. 3
- dnf.sh → /root/dnf.sh ausführen. S. 3
- dnf-automatic S. 4
- firewall-cmd S. 5
- DNS S.
- Webalizer auf Centos 8 S. 8

## Apache

- Apache installieren und konfigurieren
  - o Mod\_pagespeed (Google SEO)
  - o php-fpm
  - o Mod\_security (hardcore)
- Vsftpd konfigurieren (Mit MySQL)
- Iptables Firewall → fwbuilder.sourceforge.net (FirewallBuilder)
- MySQL Server
- Tcpwrapper
- Webmin, VirtualX
- Letsencrypt / certbot
- System Start/stop, Überflüssige Dienst abschalten bringt Security und Speed
- Postfix konfigurieren
- SSH Config
- DNS PowerDNS
- User shell → /bin/false (FTP)
- etc/skel
- fail2ban
- Quotas aktivieren (Userquota) - Siehe Artikel Aus Browser Lesezeichen Ordner: SERVER

- Webmin „eigene Login Seite“
- Vsftpd -> hide files/folders
- NTP
- Dovecot -> MySQL
- Vacation
- Spamassassin

## DNS

- Einleitung

S. 9

# Start

Zuerst wird „Cent OS 8.3 Minimal“ von einem USB-Stick installiert. Nachdem das System installiert und die Netzwerkkarte eingerichtet, rufen wir **dnf -y update** auf, um das System auf den neuesten Stand zu bringen.

```
Infos zur Installation via USB-Stick:  
https://wiki.centos.org/HowTos/InstallFromUSBkey
```

Damit wir alle notwendigen Pakete erhalten, installiere ich das EpeL Release und die yum-utils.

```
$ dnf -y install epel-release yum-utils
```

Es wird empfohlen auch die PowerTools zu aktivieren, seit EPEL Pakete Dependencies von den PowerTools hat.

```
$ dnf config-manager --set-enabled PowerTools  
or  
$ dnf config-manager --set-enabled powertools
```

Nun installiere ich alle benötigten Pakete via dnf nach. (Script: dnh.sh). Der Schalter -y bewirkt dabei, dass alle Fragen automatisch mit Ja beantwortet werden.

```
#!/bin/bash  
  
dnf -y install httpd mysql mysql-server webalizer php php-mysqlnd net-  
tools which fail2ban certbot bind-utils whois postfix postfix-mysql figlet  
php-json mod_fcgid vsftpd php-mbstring dovecot dovecot-mysql rsyslog  
iptraf-ng dnf-automatic perl-DBI perl-DBD-MySQL gd gd-devel php-gd perl-  
Net-SSLeay python3-certbot-apache spamassassin tcp_wrappers php-soap php-  
xml mod_http2 at conntrack-tools rsync tar wget
```

Wurden alle Pakete installiert müssen einige Dienste bei Start des Servers gestartet werden.

## Sendmail/ atd Prework

Da ich Postifx benutzte, CentOt 8 aber immer noch sendmail als Standardmailer definiert hat, muss zuerst sendmail sauber abgeschaltet und deaktiviert werden. Ausserdem schalte ich noch atd ab

```
$ systemctl disable sendmail
$ systemctl disable atd
$ systemctl stop sendmail
$ systemctl stop atd
```

```
$ systemctl enable httpd
$ systemctl enable php-fpm
$ systemctl enable mysqld
$ systemctl enable fail2ban
$ systemctl enable postfix
$ systemctl enable vsftpd
$ systemctl enable dovecot
```

```
$ systemctl start httpd
$ systemctl start php-fpm
$ systemctl start mysqld
$ systemctl start fail2ban
$ systemctl start postfix
$ systemctl start vsftpd
$ systemctl start dovecot
```

# dnf-automatic

Damit wir später Security Updates automatisch erhalten, nutzen wir dnf automatic update für Security Patches.

```
Datei: /etc/dnf/automatic.conf  
  
upgrade_type = security  
download_updates = yes  
apply_updates = yes  
system_name = webserver
```

Nächster Schritt, dnf-automatic aktivieren und starten

```
systemctl enable --now dnf-automatic.timer
```

Status Check

```
systemctl list-timers *dnf-*
```

Weiterführende Information:

```
https://www.tecmint.com/dnf-automatic-install-security-updates-automatically-in-centos-8/
```

# Firewall-CMD

Damit die Dienste von aussen erreichbar sind, muss die Firewall von Cent-OS angepasst werden. CentOS nutzt, wie viele andere Distributionen den Befehl firewall-cmd um Firewall Regeln lokal zu speichern. Zum Zeitpunkt der Erstellung dieser Anleitung, ist die Funktion AllowZoneDrifting deprecated. Es ist als sinnvoll die Einstellung direkt abzuschalten.

```
Datei: /etc/firewalld/firewalld.conf
```

```
AllowZoneDrifting=no
```

Um Dienste in die Firewall einzutragen trage die folgenden Befehle ein:

```
$ firewall-cmd --zone=public --add-service=ssh --permanent
$ firewall-cmd --zone=public --add-service=ftp --permanent
$ firewall-cmd --zone=public --add-service=http --permanent
$ firewall-cmd --zone=public --add-service=https --permanent
$ firewall-cmd --zone=public --add-service=smtp --permanent
$ firewall-cmd --zone=public --add-service=pop3s --permanent
$ firewall-cmd --zone=public --add-service=imaps --permanent
$ firewall-cmd --zone=public --add-port=88/tcp --permanent
$ firewall-cmd --zone=public --add-port=587/tcp --permanent
```

```
$ firewall-cmd --reload
```

Weiterführende Informationen:

```
https://www.digitalocean.com/community/tutorials/how-to-set-up-a-firewall-using-firewalld-on-centos-8-de
```

Den Dienst ,cockpit' schalten wir in der Firewall direkt ab.

```
$ firewall-cmd --zone=public --remove-service=cockpit --permanent
```

Wenn es Probleme mit dem dbus gibt (Gerne auf Virtuellen Servern) und du den folgenden Fehler erhältst:

ERROR:dbus.proxies:Introspect error on :1.23:/org/fedoraproject/FirewallD1:  
dbus.exceptions.DBusException: org.freedesktop.DBus.Error.NoReply: Did not receive a reply. Possible causes include: the remote application did not send a reply, the message bus security policy blocked the reply, the reply timeout expired, or the network connection was broken.

Gibt es folgendes Workaround für dich:

```
systemctl stop firewalld  
firewall-offline-cmd --add-service=https  
firewall-offline-cmd --add-service=http  
firewall-offline-cmd --add-service=ssh  
firewall-offline-cmd --add-service=ftp  
firewall-offline-cmd --add-service=pop3s  
firewall-offline-cmd --add-service=imaps  
firewall-offline-cmd --add-port=88/tcp  
firewall-offline-cmd --add-port=587/tcp  
systemctl start firewalld
```

Quelle: <https://help.ssh.com/support/solutions/articles/36000219298-error-dbus-proxies-introspect-error-during-postinstall>



# Apache

Grundkonfiguration eines Apache Webservers als virtuelle Schleudermaschine. Wir wollen viele Webs auf dem Server hosten. Dementsprechend wird der Apache angepasst.

## DNS (Power DNS 4.3)

Für optimalen Support sollte die Datenbank ,Transactions' unterstützen. PowerDNS wird auch ohne Transactions laufen. Jedoch können Einträge beschädigt werden (Bei einem Zonentransfer). Um Transactions nutzen zu können, muss die Datenbank auf **InnoDB** gesetzt werden.

### Installation

```
dny -y install pdns pdns-recursor pdns-tools pdns-backend-mysql
```

Die Config Datei befindet sich in /etc/pdns/pdns.conf

Dokumentation:

```
https://doc.powerdns.com/authoritative/backends/generic-mysql.html  
https://computingforgeeks.com/install-powerdns-on-centos-with-powerdns-admin/
```

MySQL root Passwort auf der Console setzen:

```
mysqladmin -u root password 'dein-neues-passwort'
```

To activate Mysql-Support change ,launch' in /etc/pdns/pdns.conf to:  
launch=gmysql

## Default Schema 4.3

```
CREATE TABLE domains (  
  id                INT AUTO_INCREMENT,  
  name              VARCHAR(255) NOT NULL,  
  master            VARCHAR(128) DEFAULT NULL,  
  last_check        INT DEFAULT NULL,  
  type              VARCHAR(6) NOT NULL,  
  notified_serial    INT UNSIGNED DEFAULT NULL,  
  account            VARCHAR(40) CHARACTER SET 'utf8' DEFAULT NULL,  
  PRIMARY KEY (id)  
) Engine=InnoDB CHARACTER SET 'latin1';  
  
CREATE UNIQUE INDEX name_index ON domains(name);  
  
CREATE TABLE records (  
  id                BIGINT AUTO_INCREMENT,  
  domain_id         INT DEFAULT NULL,  
  name              VARCHAR(255) DEFAULT NULL,  
  type              VARCHAR(10) DEFAULT NULL,  
  content            VARCHAR(64000) DEFAULT NULL,  
  ttl               INT DEFAULT NULL,  
  prio              INT DEFAULT NULL,  
  disabled           TINYINT(1) DEFAULT 0,  
  ordername         VARCHAR(255) BINARY DEFAULT NULL,  
  auth              TINYINT(1) DEFAULT 1,  
  PRIMARY KEY (id)  
) Engine=InnoDB CHARACTER SET 'latin1';  
  
CREATE INDEX nametype_index ON records(name,type);  
CREATE INDEX domain_id ON records(domain_id);  
CREATE INDEX ordername ON records (ordername);
```

```

CREATE TABLE supermasters (
    ip                VARCHAR(64) NOT NULL,
    nameserver        VARCHAR(255) NOT NULL,
    account            VARCHAR(40) CHARACTER SET 'utf8' NOT NULL,
    PRIMARY KEY (ip, nameserver)
) Engine=InnoDB CHARACTER SET 'latin1';

CREATE TABLE comments (
    id                INT AUTO_INCREMENT,
    domain_id         INT NOT NULL,
    name              VARCHAR(255) NOT NULL,
    type              VARCHAR(10) NOT NULL,
    modified_at       INT NOT NULL,
    account            VARCHAR(40) CHARACTER SET 'utf8' DEFAULT NULL,
    comment            TEXT CHARACTER SET 'utf8' NOT NULL,
    PRIMARY KEY (id)
) Engine=InnoDB CHARACTER SET 'latin1';

CREATE INDEX comments_name_type_idx ON comments (name, type);
CREATE INDEX comments_order_idx ON comments (domain_id, modified_at);

```

```

CREATE TABLE domainmetadata (
    id                INT AUTO_INCREMENT,
    domain_id         INT NOT NULL,
    kind              VARCHAR(32),
    content            TEXT,
    PRIMARY KEY (id)
) Engine=InnoDB CHARACTER SET 'latin1';

CREATE INDEX domainmetadata_idx ON domainmetadata (domain_id, kind);

CREATE TABLE cryptokeys (
    id                INT AUTO_INCREMENT,
    domain_id         INT NOT NULL,
    flags             INT NOT NULL,
    active            BOOL,
    published         BOOL DEFAULT 1,
    content            TEXT,
    PRIMARY KEY(id)
) Engine=InnoDB CHARACTER SET 'latin1';

CREATE INDEX domainidindex ON cryptokeys(domain_id);

CREATE TABLE tsigkeys (
    id                INT AUTO_INCREMENT,
    name              VARCHAR(255),
    algorithm          VARCHAR(50),
    secret            VARCHAR(255),
    PRIMARY KEY (id)
) Engine=InnoDB CHARACTER SET 'latin1';

CREATE UNIQUE INDEX namealgoindex ON tsigkeys(name, algorithm);

```

## Default Config /etc/pdns/pdns.conf

```
launch=gmysql
local-address=192.168.1.1
master=yes

#####

# MySQL Support

#####

gmysql-socket=/var/lib/mysql/mysql.sock

gmysql-dbname=powerdns

gmysql-user=***

gmysql-password=***
```

# Quota

Quota aktivieren

Via webmin mit Suche nach „Festplatten Kontigente“ oder <https://webmin:88/quota>

Benutze Quoten? „Nur Benutzer“

Oder via terminal.

Die Festplatte muss auf Quoten vorbereitet werden, dies geschieht in der Datei

```
/etc/fstab
```

Ich möchte nur Usrquota aktivieren (keine gruppenquota)

In der Datei suche ich mir die Festplatte raus, welche Quoten unterstützen soll.

In meinem Fall ist das:

```
/dev/mapper/cl-home /home xfs defaults 0 0
```

Hier füge ich an defaults den Befehl usrquota mit einem Kommata an.

```
/dev/mapper/cl-home /home xfs defaults,usrquota 0 0
```

Im Anschluss muss die Festplatte neu gemountet werden.

```
Mount -o remount /home
```

<https://www.linuxtechi.com/enable-user-group-disk-quota-on-centos-7-rhel-7/>

# VSFTPD

VSFTPD User sollten aus Sicherheitsgründen keine Shell erhalten. Der User sollte mit `/bin/false` oder `/sbin/nologin` erstellt werden. Damit wir User ohne Shell an den vsftpd Server einloggen können, muss in `/etc/pam.d/vsftpd` folgende Zeile kommentiert werden.

```
auth required pam_shells.so
```

```
##PAM-1.0
session optional pam_keyinit.so force revoke
auth required pam_listfile.so item=user sense=deny file=/etc/vsftpd/ftpusers onerr=succeed
#auth required pam_shells.so
auth include password-auth
account include password-auth
session required pam_loginuid.so
session include password-auth
```

*Quelle:*

<https://linux-tips.com/t/users-with-bin-false-shell-to-login-on-vsftpd/200>

Chroot local users:

Um User lokal im Homeverzeichnis zu chrooten

# BitWorker Stuff

```
chroot_local_user=YES
hide_file={*.virtualx,.bash_logout,.bash_profile,.bashrc,WEBSTATS}
force_dot_files=YES
```

Der übergeordnete Ordner darf nur Rechte 551 haben ( also `/home/httpd/www.domainnam.de` )



## SSL Konfig

```
ssl_enable=YES
allow_anon_ssl=NO
force_local_data_ssl=YES
force_local_logins_ssl=YES
ssl_tlsv1_1=YES
ssl_tlsv1_2=YES
ssl_tlsv1=NO
ssl_sslv2=NO
ssl_sslv3=NO
require_ssl_reuse=YES
ssl_ciphers=HIGH
rsa_cert_file=/etc/letsencrypt/live/srv.bit-worker.com/fullchain.pem
rsa_private_key_file=/etc/letsencrypt/live/srv.bit-worker.com/privkey.pem
```

## Pam\_mysql für vsftpd

Das Programm pam\_mysql gibt es hier.

Binary Quelle: [https://centos.pkgs.org/8/cheese-x86\\_64/pam\\_mysql-0.8.1-0.6.el8.x86\\_64.rpm.html](https://centos.pkgs.org/8/cheese-x86_64/pam_mysql-0.8.1-0.6.el8.x86_64.rpm.html)

HowTo Quelle: <https://www.linux.com/news/set-virtual-ftp-server-pam-mysql>

In die Datei /etc/pam.d/vsftpd folgende Zeilen zufügen:

```
auth required /lib64/security/pam_mysql.so user=root passwd=DEINPASSWORT host=localhost db=virtualx table=passwd
usercolumn=username passwdcolumn=passwd crypt=3 sqllog=1 logtable=logs logmsgcolumn=msg logusercolumn=user
logpidcolumn=pid loghostcolumn=host logrhostcolumn=rhost logtimecolumn=logtime

account required /lib64/security/pam_mysql.so user=root passwd=DEINPASSWORT host=localhost db=virtualx table=passwd
usercolumn=username passwdcolumn=passwd crypt=3 sqllog=1 logtable=logs logmsgcolumn=msg logusercolumn=user
logpidcolumn=pid loghostcolumn=host logrhostcolumn=rhost logtimecolumn=logtime
```

Gute PAM Mysql Doku Quellen:

[https://wiki.asta.hs-fulda.org/images/7/7e/Doku\\_pamnssmysql.pdf](https://wiki.asta.hs-fulda.org/images/7/7e/Doku_pamnssmysql.pdf)

<https://github.com/NigelCunningham/pam-MySQL>

Die 3 vorhandenen auth Module in /etc/pam.d/vsftpd werden kommentiert:

```
#auth    required    pam_listfile.so item=user sense=deny file=/etc/vsftpd/ftpusers onerr=succeed
#auth    required    pam_shells.so
#auth    include     password-auth
```

In /etc/vsftpd/vsftpd.conf:

```
pam_service_name=vsftpd
```

ErstelleOrder:

```
/etc/vsftpd/vsftpd_user_conf
```

Usefull um MD5 zu kontrollieren: →<https://www.md5-generator.de/>

Tipp:

pam\_mysql kann mit crypt=2 oder crypt=3 in/etc/pamd./vsftpd konfiguriert werden

Crypt=2 ist dabei die SQL Methode Mysql PASSWORD'meinpass')

Crypt=3 ist die md5 Methode (Virtual x produziert in /,pduls/add\_user.pl Zeile 136ff ein md5 passwort

Um durch eine iptables Firewall per „FTP over TLS“ durchzukommen, muss der „Passiv Mode“ in Vsftpd eingeschaltet werden. Hier eine Beispielkonfiguration. Die Ports sind natürlich frei wählbar. Sollten dann aber in der iptables Firewall eingetragen sein.

```
pasv_enable=YES
pasv_min_port=10000
pasv_max_port=11000
```

# CERTBOT

Ein Zertifikat zufügen. (Die Domain muss bereits per http, also Port 80 erreichbar sein)

```
certbot certonly -d srv.bit-worker.com
```

<https://www.howtoforge.de/anleitung/wie-man-lets-encrypt-ssl-tls-zertifikate-mit-certbot-erstellt-und-verwaltet/>

<https://www.teslina.com/tutorials/sicherheit/ssl/lets-encrypt-ssl-zertifikate-mit-certbot-erstellen/>

Teststuff:

```
certbot --apache certonly -n -d srv.bit-worker.com
```

## IMPORTANT NOTES:

- Congratulations! Your certificate and chain have been saved at:  
/etc/letsencrypt/live/srv.bit-worker.com/fullchain.pem
- Your key file has been saved at:  
/etc/letsencrypt/live/srv.bit-worker.com/privkey.pem

Certbot renew /etc/cron.weekly

```
#!/bin/bash  
certbot renew  
exit
```

Auf der Shell kann man das Apache Modul wie folgt benutzen

certbot –apache

Infos zum .timer hier:

<https://www.digitalocean.com/community/tutorials/how-to-secure-apache-with-let-s-encrypt-on-ubuntu-20-04-de>

# SSH

Zur Erhöhung der Sicherheit sollte der SSH Dienst angepasst werden. Ich habe PermitRootLogin und PasswordAuthentication in der /etc/sshd\_config auf no gesetzt bzw. einkommentiert.

Anschließend eine Match Host basierende Version implementiert:

```
PubkeyAuthentication no
PermitRootLogin no
PasswordAuthentication no

Match Host <meine ip adresse>
    PasswordAuthentication yes
    PermitRootLogin yes
```

# Fail2ban

Fail2ban muss in CentOS 8.3 in der Grundkonfiguration angepasst werden. In der Standard Konfiguration wird iptables als Block Maschine verwendet. Iptables ist aber in Cent OS 8 nicht mehr die erste Wahl. Stattdessen setzen die Jungs auf den firewalld. Ich habe die Änderung an der sshd Jail gelistet. Der Knackpunkt ist die banaction. Sie kann natürlich auch gleich im Standard Bereich von jail.conf angepasst werden, dann gilt sie automatisch für alle jails.

/etc/fail2ban/jail.conf

```
[sshd]

port = ssh
logpath = %(sshd_log)s
backend = %(sshd_backend)s
enabled = true
bantime = 1h
maxretry = 1
banaction = firewallcmd-allports
```

# Postfix

Postfix mit Dovecot SASL Support - SASL installieren

/etc/postfix/main.cf

```
### sasl howto

smtpd_sasl_auth_enable = yes
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_tls_auth_only = yes
broken_sasl_auth_clients = yes
smtpd_sasl_security_options = noanonymous
```

## Virtual Mailboxes, relay\_restrictions, recipient\_restrictions in /etc/main.cf

```
virtual_mailbox_base = /virtualmail/
virtual_mailbox_domains = mysql:/etc/postfix/mysql-domains.cf
virtual_maps = mysql:/etc/postfix/mysql-virtual.cf

smtpd_relay_restrictions =
    permit_sasl_authenticated
    permit_mynetworks
    reject_unauth_destination

smtpd_recipient_restrictions =
    check_recipient_access hash:/etc/postfix/access,
    check_sender_access hash:/etc/postfix/access,
    permit_sasl_authenticated,
    permit_mynetworks,
    reject_non_fqdn_recipient,
    reject_unknown_recipient_domain,
    reject_unknown_sender_domain,
    reject_unauth_destination,
    reject_unauth_pipelining,
    reject_invalid_hostname,
    reject_rbl_client relays.ordb.org,
    reject_rbl_client bl.spamcop.net,
    reject_rbl_client sbl.spamhaus.org,
    reject_rbl_client dsn.rfc-ignorant.org,
    reject_rbl_client cbl.abuseat.org,
    reject_rbl_client opm.blitzed.org
```



/etc/postfix/master.cf

```
smtp    inet  n       -       n       -       -       smtpd
#smtp   inet  n       -       n       -       1       postscreen
#smtpd  pass  -       -       n       -       -       smtpd
#dnsblog unix -       -       n       -       0       dnsblog
#tlsproxy unix -       -       n       -       0       tlsproxya

# port 587
submission inet n       -       n       -       -       smtpd
  -o syslog_name=postfix/submission
  -o smtpd_tls_security_level=encrypt
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_tls_auth_only=yes
# -o smtpd_reject_unlisted_recipient=no
# -o smtpd_client_restrictions=$mua_client_restrictions
# -o smtpd_helo_restrictions=$mua_helo_restrictions
# -o smtpd_sender_restrictions=$mua_sender_restrictions
# -o smtpd_recipient_restrictions=
  -o smtpd_relay_restrictions=permit_sasl_authenticated,reject
# -o milter_macro_daemon_name=ORIGINATING
```

\$ postfix reload

```
systemctl start saslauthd
systemctl enable saslauthd
systemctl restart postfix
```

# Dovecot

Dovecot MySQL Support und nur auf den Ports 993 und 995 (imps und pop3s mit TLS)

Vorarbeiten:

Zuerst bitte kontrollieren das das Paket „dovecot-mysql“ installiert ist.

Ordner: „/home/pop/“ muss chmod 777 sein.

In der /etc/dovecot/dovecot.conf Datei den Standard wechseln (Siehe Beispiel)

```
#protocols = imap pop3 lmtp submission
protocols = imap pop3
```

File /etc/dovecot/10-master.conf

```
inet_listener imap {
  #port = 143
  port = 0
}

service pop3-login {
  inet_listener pop3 {
    #port = 110
    port = 0
  }
}
```

In der /etc/dovecot/conf.d/10-ssl.conf die Zertifikate anpassen an die eigenen

```
#ssl_cert = </etc/pki/dovecot/certs/dovecot.pem
#ssl_key = </etc/pki/dovecot/private/dovecot.pem

ssl_cert = </etc/letsencrypt/live/srv.bit-worker.com/fullchain.pem
ssl_key = </etc/letsencrypt/live/srv.bit-worker.com/privkey.pem
```

/etc/dovecot/conf.d/10-mail.conf

```
mail_location = mbox:/home/pop/%u
```

Better use maildir instead of mbox:

```
mail_location = maildir:/home/pop/%u
```

Aktivieren von Dovecot SQL Auth

- 1.) In Datei 10-auth.conf !include auth-sql.conf.ext auskommentieren
- 2.) Datei „/etc/dovecot/dovecot-sql.conf.ext“ erstellen.

Usernamen umschreiben. Um es dem Kunden einfacher zu machen schreibe ich mail1@domainname.de um in mail1.domainname.de. Viele Kunden melden sich mit der E-Mail an und damit das funktioniert in dieser Datei die Änderung wie folgt vornehmen.

/etc/dovecot/conf.d/10-auth.conf

```
auth_username_translation = "@."
```

**TODO:.** ISP, Thunderbird, Autoconfig

ISPDB

<https://developer.mozilla.org/en-US/docs/Mozilla/Thunderbird/Autoconfiguration>

<https://wiki.mozilla.org/Thunderbird:Autoconfiguration:ConfigFileFormat>

<https://syncserver.at/dashboard/portfolio-autoconfig.php>

## Dovecot SASL Support

/etc/dovecot/conf.d/10-auth.conf die Zeile auth-master auskommentieren:

```
!include auth-master.conf.ext
```

/etc/dovecot/conf.d/10-master.conf

```
...
service auth {
...
  # Postfix smtp-auth
  unix_listener /var/spool/postfix/private/auth {
    mode = 0666
  }
...
}
```

In /etc/dovecot/conf.d/10-auth.conf den authc mechanismus „login“ zufügen.

```
auth_mechanisms = plain login
```

Quellen:

<https://help.ubuntu.com/community/PostfixDovecotSASL> (sehr gute Dokumentation)

<https://wiki2.dovecot.org/HowTo/PostfixAndDovecotSASL>

# iptables

Seit RHEL8 / CentOS8 ist firewalld im Einsatz um iptables zu kontrollieren. Ich mag das ganze firewalld Konzept vorne und hinten nicht. Darüber hinaus habe ich immer wieder Probleme mit dem dbus, der dann nichts in die Firewall eintragen will. (Z.B. geht fail2ban mit firewalld gar nicht) Ich gehe also zurück zu den roots und deaktiviere den firewalld Daemon.

```
$ systemctl stop firewalld  
$ systemctl disable firewalld
```

Erstelle ein neues Start/Stop Script

```
[Unit]  
Description=BitWorker Firewall  
After=network.service  
  
[Service]  
  
ExecStart=/etc/firewall/srv.bit-worker.com.fw  
#ExecStop=/etc/firewall/reset.sh  
#PIDFile=/etc/firewall/firewall.pid  
  
[Install]  
WantedBy=multi-user.target
```

Starten und beim Hochfahren aktivieren

```
$ systemctl start firewall.service  
$ systemctl enable firewall.service
```