# CentOS 7

How to install a webserver on CentOS7 from scratch

- Install Apache, MariaDB, PHP73 on CentOS 7
- Get Certificates for Webserver
- Yum
- iptables - block ipnummer
- awstats

# Install Apache, MariaDB, PHP73 on CentOS 7

## Installation done on OVH VPS (last time augustus 2019)

### Add normal user to work under

```
useradd max
passwd max
vi /etc/sudoers
max ALL=(ALL:ALL) ALL
```

### ssh, change port for security

```
vi /etc/ssh/sshd_config (change port 22 to 1122)
service sshd restart
semanage port -a -t ssh_port_t -p tcp 1122
sudo firewall-cmd --list-all
sudo firewall-cmd --permanent --zone=public --add-port=1122/tcp
```

### Install MariaDB

```
sudo yum -y install mariadb-server mariadb
sudo systemctl start mariadb.service
sudo systemctl enable mariadb.service
mysql_secure_installation
```

### Install Apache

```
sudo yum -y install httpd
sudo systemctl start httpd.service
sudo systemctl enable httpd.service
```

### Cinfigure Firewall

```
sudo firewall-cmd --permanent --zone=public --add-service=http
sudo firewall-cmd --permanent --zone=public --add-service=https
sudo firewall-cmd --reload
```

**Install PHP7.3**

```
sudo rpm -Uvh http://rpms.remirepo.net/enterprise/remi-release-7.rpm
sudo yum -y install yum-utils
sudo yum update
sudo yum-config-manager --enable remi-php73
```

Optional, fix issue ‘Loaded plugins: fastestmirror’; this message appears when installing PHP7.3 and installation is not executed.

```
sudo /etc/yum/pluginconf.d/fastestmirror.conf
and change enabled=1 -> enabled=0
sudo yum -y install php php-opcache
sudo systemctl restart httpd.service
```

Add max (user) to groups

```
sudo usermod -a -G root max
sudo usermod -a -G apache max
```

Install Python plus Libraries for maxtrack (runtracker)

```
sudo yum -y install python-pip
pip install flask
pip install stravalib
pip install TinyDB
```

xxx

xxx

# Get Certificates for Webserver

## Use letsencrypt

https ssl install

Instruction from https://certbot.eff.org/lets-encrypt/centosrhel7-apache

Once installed, **add** certificate with: `sudo certbot --apache`

**Renew** certificates: `sudo certbot renew`

**crontab** (auto renew) `sudo crontab -e`

```
# Minute   Hour   Day of Month       Month          Day of Week        Command
# (0-59)  (0-23)     (1-31)    (1-12 or Jan-Dec)  (0-6 or Sun-Sat)
1       7       *       *       1       /usr/bin/certbot renew --quiet
```

# Ubuntu

## DNS

```
A Record: @ 12.13.14.15
CNAME Record: www domain.com
```

## vhost

Rewrite Engine added by certbot

```
<VirtualHost>
ServerName domain.com
ServerAlias www.domain.com
DocumentRoot /var/www/domain
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
RewriteEngine on
RewriteCond %{SERVER_NAME} =www.domain.com [OR]
RewriteCond %{SERVER_NAME} =domain.com
RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI} [END,NE,R=permanent]
```

```
</VirtualHost>
```

## certbot

Eén domein heeft meerdere sub-domeinen: meldt ze in één keer aan.

Komt er één bij dan de complete regel opnieuw invoeren.

```
sudo certbot --apache -d domain.com -d www.domain.com
```

## certbot - other

```
// delete
sudo certbot delete

//list
sudo certbot certificates
```

--

# Yum

Belangrijkste yum commando's

```
# show installed packages
yum list installed

# Check for updates
yum check-update

# Repolist
yum repolist

# update
sudo yum update
```

# iptables - block ipnummer

*iptables is de (software) firewall van (o.m.) CentOS. In dit stukje leer je hoe je ipnummers kunt blokkeren met iptables.*

Check op invalid logins. Meestal ssh

```
sudo grep failed /var/log/audit/audit.log* | grep -E -o "([0-9]{1,3}[\.]){3}[0-9]{1,3}" | sort | uniq -c
```

Dit laat een lijstje zien van ip nummers waarvandaan failed login's zijn geregstreerd. Het getal voor het ip nummer is het aantal keren dat er een failed login heeft plaatsgevonden.

Stel ipnummer 45.119.53.58 komt meer dan 1000 keer voor. Dan kun je eerst proberen op te zoeken waar dit nummer vandaan komt. Daar zijn verschillende sites (bijvoorbeeld http://whois.domaintools.com ) voor en je kunt zelf ook databases downloaden. In dit geval komt dit nummer uit China.

Nu kun je dit ipnummer blokkeren. Beter nog is om heel het netwerk te blokeren. Meestal kun je als subnetmask /24 nemen daarmee blokkeer je 256 ip adressen. In dit geval blokkeer je dan 45.119.53.*

Je kunt ook nog meer blokkeren, bijvoobeeld het /16 netwerk oftewel 45.118.* dan blokkeer je 65 536 (64K) ip adressen. Maar dan moet je wel weten welke netwerken je dan allemaal blokkeert. Om dit te controleren heb je een uitgebreide datasabase nodig, die je kunt downloaden bij bijvoorbeeld https://lite.ip2location.com Op deze site kun je zelf ook een een lijst krijgen van bijvoorbeeld alle ipnummers uit Nederland. Dan kun je in ieder geval controleren of je geen Nederlandse ipnummers blokkeert.

Blokkeren zelf gebeurt met:

```
sudo iptables -A INPUT -s 45.119.53.0/24 -j DROP
```

In dit geval blokkeer je 45.119.53.*

Om te zien wat je hebt geblokeerd:

```
sudo iptables -L
```

Om alle blokkades op te heffen:

```
iptables -F INPUT
```

# awstats

## stats

## Install

Ubuntu: https://tecadmin.net/install-awstats-apache-log-analyzer-on-ubuntu/

CentOS: https://tecadmin.net/steps-to-configure-awstats-on-centos-and-rhel-system/

## Add site

1. ga naar /etc/awstats
2. create new file awstats.xxx.conf waarbij xxx de domeinnaam is, bijvoorbeeld awstats.roc.ovh.conf
3. pas file aan, pas de domainnaam aan en pas de naam van de logfile die moet worden ingelezen aan.
4. pas de awstats-update.sh file aan en zet de nieuwe domeinnaam in de eerste regel van de file (spreekt voor zich).
5. edit vi /var/www/default/stats/index.php en voeg de site toe (voor de web toegang)

Vanuit de crontab wordt de awstats-update.sh file elke nacht gedraaid om 01:00 uur.

Let op dat de user waarvan uit de crontab wordt gedraaid toegang heeft tot de logfiles.

Test de awstats-update.sh door deze handmatig te draaien.

## File locaties

output data files: /var/lib/awstats
cronjob: /etc/awstats/awstats-update.sh

### config file

```
# /etc/awstats/awstats.mijnsite.com.conf
```

```
LogFile="/var/log/apache2/mijnsite-access.log"
SiteDomain="mijnsite.com"
HostAliases="mijnsite.com www.mijnsite.com"
```

## awstats-update.sh

```
for i in mijnsite.com anothersite.com andanothersite.com; do
        echo "------------------------------------------------------"
        echo "    Udate stats for:"$i
        echo "------------------------------------------------------"
        perl /usr/lib/cgi-bin/awstats.pl -config=$i -update
done

# filter url from stats
find /var/lib/awstats/*.txt -exec sed -i 's/phpmyadmin\.717664/phpmyadmin-local/g' {} \;
```

## crontab

```
0 2 * * * /etc/awstats/awstats-update.sh > /home/.../awstats/awstats-lastrun.log
```