# Server Installationsanleitung

## Betriebssystem Installation

* Das Image von raspberry.com herunterladen und extrahieren. (<https://www.raspberrypi.org/downloads/raspbian/>)
* Um die Image auf der SD Karte zu bringen benutzt man Etcher. (<https://etcher.io/>)
* Mit den Standard-Anmeldedaten Anmelden. User: pi Password: raspberry
* System aktualisieren: sudo apt-get update & sudo apt-get upgrade
* SSH aktivieren:
  + sudo raspi-config in ein Terminal eingeben.
  + Interfacing Options auswählen.
  + SSH auswählen -> aktivieren.

## Software Installation

Nun werden die benötigten Dienste und Tools installiert, die von dem Server benötigt werden.

* Installation der Webserver. (PHP, Nginx, MySQL, Java SDK, Composer, Utils)

apt-get install nginx

udo apt-get install mysql-server

apt-get install php5-fpm php5-mysql

sudo apt-get install mysql-server mysql-client

sudo apt-get install oracle-java8-jdk  
sudo apt-get install curl php5-cli git  
curl -sS https://getcomposer.org/installer | sudo php -- --install-dir=/usr/local/bin --filename=composer

## Erstellung SSL Zertifikate

Bevor die Webapplikationen installiert werden können, müssen die Zertifikate generiert werden. (Self-Signed)

* Erstellung SSL Zertifikat für die Client Webapplikation.  
  Als Hostname wird hier als Beispiel intercom.app verwendet. Zuerst muss eine Konfigurationsdatei *(v3.ext)* mit dem folgenden Inhalt generiert werden:

authorityKeyIdentifier=keyid,issuer

basicConstraints=CA:TRUE

keyUsage = digitalSignature, nonRepudiation, keyEncipherment, dataEncipherment

subjectAltName = @alt\_names

[alt\_names]

DNS.1 = intercom.app

Falls ein IP Als hostname verwendet wird, kann man *@alt\_names* mit *IP:192.168.0.18* ersetzen.

* Nun müssen folgende Befehle eingegeben werden. Wenn gefragt, muss der Hostname oder IP als Common Name (CN) Eingegeben werden. Als Password kann immer dieselbe verwendet werden und muss das Keystore Password der Signaling-Server entsprechen.

sudo openssl genrsa -des3 -out rootCA.key 2048

sudo openssl req -x509 -new -nodes -key rootCA.key -sha256 -days 1024 -out rootCA.pem

sudo openssl req -new -sha256 -nodes -out server.csr -newkey rsa:2048 -keyout server.key

sudo openssl x509 -req -in server.csr -CA rootCA.pem -CAkey rootCA.key -CAcreateserial -out server.crt -days 500 -sha256 -extfile v3.ext

sudo openssl pkcs12 -export -in server.crt -inkey server.key -out cert.p12

sudo keytool -importkeystore -srckeystore cert.p12 -srcstoretype PKCS12 -destkeystore keystore.jks -deststoretype JKS

sudo openssl x509 -inform PEM -outform DER -in server.crt -out phone.der.crt

Somit wurden diverse Dateien generiert. Die folgenden werden später gebraucht.

* + server.cert & server.key -> SSL Zertifikate für Apache2
  + rootCA.pem -> Root CA. Das muss in den Client-Browser importiert werden, damit die Clients den Server als Vertraulich erkennen.
  + phone.der.crt -> Root CA für Mobilegeräte. Bei Mobilegeräte kann es per E-Mail verschickt werden und dann aus der Systemeinstellungen installiert werden.
  + keystore.jks -> Das muss später in die Selber Verzeichnis kopiert werden, wo der SignalingServer installiert wird.

## Konfiguration von Nginx

Vor dem Deploy den Webapplikationen muss der Webserver noch konfiguriert werden.

* In der Datei /etc/php5/fpm/php.ini muss die folgende Zeile auskommentiert und editiert werden:

cgi.fix\_pathinfo=0

* Nun muss Nginx so konfiguriert werden, dass PHP als compiler verwendet wird. Der Datei /etc/nginx/sites-available/default muss editiert werden. Im rot markiert sind die Stellen die angepasst werden müssen.
* Nun müssen die zwei VirtualHosts für die zwei WebApps konfiguriert werden. Diese werden unter verschiedene Ports laufen. Für das müssen zwei Konfigurationsdatei unter /etc/nginx/sites-available erstellt werden*. Bsp: intercom.app & management.app.*

Der Inhalt muss wie folgendes aussehen. Im rot markiert sind die Stellen, die unterschiedlich sein müssen für die beiden Webapplikationen. Der Path zu den SSL-Zertifikate muss auch angepasst werden.

# Default server configuration

server {

# SSL configuration

listen 443 ssl; # 444 for the second host

listen [::]:443 ssl;

ssl\_certificate /path/to/the/certificate/server.crt;

ssl\_certificate\_key /path/to/the/certificate/server.key;

root /var/www/intercom/public; # /management/public for the second host

# Add index.php to the list if you are using PHP

index index.php index.html index.htm index.nginx-debian.html;

server\_name \_;

location / {

# First attempt to serve request as file, then

# as directory, then fall back to displaying a 404.

try\_files $uri $uri/ /index.php?$query\_string;

}

location ~ \.php$ {

include snippets/fastcgi-php.conf;

fastcgi\_pass unix:/var/run/php5-fpm.sock;

}

location ~ /\.ht {

deny all;

}

}

* Zum Abschliessen noch die folgenden Befehle eingeben:

sudo ln -s /etc/nginx/sites-available/intercom.app /etc/nginx/sites-enabled/

sud ln -s /etc/nginx/sites-available/management.app /etc/nginx/sites-enabled/

sudo service nginx reload

sudo service nginx restart

sudo service php5-fpm restart

sudo reboot

## Deploy Webapplikationen

* Die beide Webapplikationen müssen zuerst auf dem Server unter den passenden Verzeichnissen kopiert werden.
  + /var/www/management
  + /var/www/intercom
* Rechte anpassen

sudo chmod -R 775 /var/www

sudo chmod -R 777 /var/www/management/storage

sudo chmod -R 775 /var/www/intercom/storage

sudo chgrp -R www-data /var/www/

* MySQL Database erstellen, dann Anmeldedaten und Database Name in der Datei: .env eingeben. (Falls .env nicht vorhanden: cp .env.example .env)
* Webapplikation installieren: (Diese befehle müssen in der Root Dir von jede Webapp eingegeben werden.)

composer install

php artisan migrate (Muss nur für das Management Webapp gemacht werden)

php artisan key:generate

Die zwei Webapps müssten nun unter die ports 443 und 444 aufrufbar sein.

## Deploy Dienste & Services

* Zuerst die benötigten Pfade erstellen

mkdir /home/pi/server/signalingServer

mkdir /home/pi/server/relayController

* Die beide kompilierte JARs in den entsprechenden Verzeichnissen kopieren. Die kompilierten JARs sind in den jeweilige Projekts Verzeichnisse unter /deploy zu finden.
* Nun muss noch für den SignalingServer noch das vorher hergestellte keystore.jks kopiert werden. Der Keystore muss sich in dem gleichen Verzeichnis wie der Signaling Server befinden.
* Für beide Dienste muss noch das autostart Skript unter /etc/init.d/ kopiert werden. Die Skripte sind unter dem Script-Verzeichnis gespeichert. Auch diese Skripte müssen ausführbar sein. Folgende Befehle müssen noch eingegeben werden:

sudo chmod +x /etc/init.d/signalingServer

sudo chmod +x /etc/init.d/relayController

sudo update-rc.d signalingServer defaults

sudo update-rc.d relayController defaults