

Einen NGINX-Webserver installieren und mit PHP und Datenbank betreiben

NGINX installieren

Der **NGINX**-Server hat den Vorteil, dass er sehr ressourcensparend und schnell arbeitet und relativ einfach zu konfigurieren ist. Viele besondere andere Serverdienste, die sich zur Bedienung eines vorhandenen Webservers bedienen, um eine Administrationsoberfläche für ihren Dienst zur Verfügung zu stellen, bauen auf den **Apache2**-Server auf. Wer allerdings nur beabsichtigt einen einfachen Webserver mit Datenbank und PHP ohne weiteren „Schnickschnack“ zu betreiben, ist mit **NGINX** gut beraten.

Installieren wir **NGINX**

```
sudo apt update
sudo apt install nginx
```

PHP, Mariadb(MySQL) und SQLite3 installieren (mit weiteren Modulen)

```
sudo apt install php php-fpm php-json php-xml php-xmlrpc mariadb-server php-mysql sqlite3 php-sqlite3
```

NGINX anpassen und konfigurieren

Die Standard-Installation passt man dann einfach hier an:

```
sudo nano /etc/nginx/sites-available/default
```

Die Beispielkonfiguration sollte so aussehen:

```
server {
    listen 80;
    server_name your_domain_or_ip;

    root /var/www/html;
    index index.php index.html index.htm;

    location / {
        try_files $uri $uri/ =404;
    }

    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/var/run/php/php7.0-fpm.sock;
    }
}
```

```
    fastcgi_pass unix:/var/run/php/php8.2-fpm.sock; # Version
entsprechend anpassen
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
    include fastcgi_params;
}

location ~ /\.ht {
    deny all;
}
}
```

Beachte: Die PHP-Version im Beispiel auf die installierte Version zu ändern. Hier **8.2**

NGINX-testen und starten

Überprüfe die Konfiguration:

```
sudo nginx -t
```

und starte den NGINX-Server neu:

```
sudo systemctl reload nginx
```