Ubuntu 18.04: Redis als PHP-Cache

```
                _.-``__ ''-._
           _.-``    `.  `_.  ''-._           Redis 4.0.9 (00000000/0) 64 bit
       .-`` .-```.  ```\/    _.,_ ''-._
      (    '      ,       .-`  | `,    )     Running in standalone mode
      |`-._`-...-` __...-.``-._|'` _.-'|     Port: 0
      |    `-._   `._    /     _.-'    |     PID: 25321
       `-._    `-._  `-./  _.-'    _.-'
      |`-._`-._    `-.__.-'    _.-'_.-'|
      |    `-._`-._        _.-'_.-'    |           http://redis.io
       `-._    `-._`-.__.-'_.-'    _.-'
      |`-._`-._    `-.__.-'    _.-'_.-'|
      |    `-._`-._        _.-'_.-'    |
       `-._    `-._`-.__.-'_.-'    _.-'
           `-._    `-.__.-'    _.-'
               `-._        _.-'
                   `-.__.-'
```

In diesem Beispiel läuft PHP 7.2 als CGI.

Installation der Pakete

```
# apt-get update
# aptitude install redis php7.2-redis
```

Konfiguration

Die Anbindung des Webservers mit PHP soll über einen Unix-Socket erfolgen. Dazu müssen ein paar Zeilen in der Konfiguration geändert werden:

```
[...]
port 0
[...]
unixsocket /var/run/redis/redis-server.sock
unixsocketperm 770
```

Der Webserver muss Zugriff auf den Socket von Redis bekommen.

```
# usermod -a -G redis www-data
```

Die Änderungen werden nach Neustart der Dienste wirksam.

```
# service redis-server restart
# service php7.2-fpm restart
# service nginx restart
```

Performance-Tuning

Beim Systemstart wollen wir ein paar Systemparameter manipulieren. Dies kann über das Skript rc.local geschehen. Die rc.local muss unter Ubuntu jedoch zuvor angelegt werden und wird ausgewertet vom Dienst rc-local.

```
# service rc-local status
● rc-local.service - /etc/rc.local Compatibility
   Loaded: loaded (/lib/systemd/system/rc-local.service; enabled-runtime;
vendor preset: enabled)
  Drop-In: /lib/systemd/system/rc-local.service.d
           └─debian.conf
   Active: active (exited) since Sat 2019-04-13 19:55:54 CEST; 1h 52min ago
     Docs: man:systemd-rc-local-generator(8)
  Process: 1366 ExecStart=/etc/rc.local start (code=exited,
status=0/SUCCESS)

Apr 13 19:55:54 xxx systemd[1]: Starting /etc/rc.local Compatibility...
Apr 13 19:55:54 xxx rc.local[1366]: net.core.somaxconn = 65535
Apr 13 19:55:54 xxx systemd[1]: Started /etc/rc.local Compatibility.
```

Das Anlegen geschieht wie folgt:

```
# vi /etc/rc.local
# chmod +x /etc/rc.local
```

Dieser Parameter muss nur einmal in die /etc/sysctl.conf eingetragen werden:

```
# echo 'vm.overcommit_memory = 1' >> /etc/sysctl.conf
```

/etc/rc.local

```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.
```

```
echo never > /sys/kernel/mm/transparent_hugepage/enabled
sysctl -w net.core.somaxconn=65535
service redis-server restart

exit 0
```

Einbindung in die PHP-Anwendung nicht vergessen

Nextcloud

/srv/www/html/nextcloud/config/config.php

```
[...]
  'filelocking.enabled' => 'true',
  'memcache.local' => '\\OC\\Memcache\\Redis',
  'memcache.locking' => '\\OC\\Memcache\\Redis',
  'redis' =>
  array (
    'host' => '/var/run/redis/redis-server.sock',
    'port' => 0,
    'timeout' => 0.0,
  ),
[...]
```

From:
https://wikinet.webby.hetzel-netz.de/ - **Sebastians IT-Wiki**

Permanent link:
https://wikinet.webby.hetzel-netz.de/ubuntu:redis-server

Last update: **2019/04/13 19:59**