

SYSINT_7.1

Verfasser: **Leonhard Stransky, 4AHIT**

Datum: **19.09.2023**

Systemintegration "High Availability und Lastverteilung" - HAProxy

Einführung:

HAProxy (High Availability Proxy) ist eine Software zur Hochverfügbarkeit und Lastverteilung

Ziele:

Aufteilung von Verbindungen auf verschiedene Server mittels HAProxy durch das Round Robin Verfahren

Aufgabenstellung:

GK Es soll HAProxy so konfiguriert werden, dass er die Last der HTTP-Anfragen an den Webserver auf zwei Server per Round Robin Prinzip verteilt. Die Übung ist mit Hilfe von 3 virtuellen Maschinen oder Containern: * ein Loadbalancer (HAProxy) * zwei Webserver (Apache) einzurichten.

Anleitung: [HAProxy](#) konfigurieren * GK Dokumentiere die Durchführung und das Ergebnis dieser Aufgabenstellung in einem Protokoll (PDF) und lade dieses hier hoch.

Commands:

```
# create network
docker network create --driver=bridge mynetwork

# create webserver
docker run -d --name apache1 --net mynetwork -p 8080:80 httpd
docker run -d --name apache2 --net mynetwork -p 8090:80 httpd

# create haproxy
docker run -d --name haproxy --net mynetwork -v "D:\OneDrive\TGM
4AHIT\SYT\SysInt\M7.1":/usr/local/etc/haproxy:ro -p 80:80 -p 8404:8404
haproxy:latest

# open bash
docker exec -it container_name_or_id /bin/bash

# find index.html file
find / -name "index.html"
# output was: /usr/local/apache2/htdocs/index.html

# change index.html file
echo "<h1>apache1</h1>" > /usr/local/apache2/htdocs/index.html
```

HAProxy Config:

```
defaults
  mode http
  timeout client 10s
  timeout connect 5s
  timeout server 10s
  timeout http-request 10s
  log global

frontend stats
  bind *:8404
  stats enable
  stats uri /
  stats refresh 10s

frontend myfrontend
  bind :80
  default_backend webservers

backend webservers
  balance roundrobin
  server s1 apache1:80 check
  server s2 apache2:80 check
```

Quellen:

[1] [www.haproxy.com](https://www.haproxy.com/blog/how-to-run-haproxy-with-docker) 2023. *haproxy* [online] Available at: <https://www.haproxy.com/blog/how-to-run-haproxy-with-docker> [Accessed 18 March 2023].