

HAProxy

- [Overview](#)
- [Sample haproxy.cfg](#)
- [Docker Setup](#)
- [Proxying a port using HAProxy](#)
- [Reference](#)

Overview

HAProxy, which stands for High Availability Proxy, is a popular open source software TCP/HTTP Load Balancer and proxying solution which can be run on Linux, Solaris, and FreeBSD. Its most common use is to improve the performance and reliability of a server environment by distributing the workload across multiple servers (e.g. web, application, database). It is used in many high-profile environments, including: GitHub, Imgur, Instagram, and Twitter.

Sample haproxy.cfg

```
# Simple configuration for an HTTP proxy listening on port 80 on all
# interfaces and forwarding requests to a single backend "servers" with a
# single server "server1" listening on 127.0.0.1:8000
global
    daemon
    maxconn 256

defaults
    mode http
    timeout connect 5000ms
    timeout client 50000ms
    timeout server 50000ms

frontend http-in
    bind *:12345
    default_backend servers

backend servers
    server server1 192.168.1.161:80 maxconn 32
    server server2 192.168.1.82:80 maxconn 32
```

Docker Setup

Sample script to build an HAProxy Container using the haproxy.cfg file copied to a subfolder named 'conf'.

buildDocker.sh

```
CONTAINER=haproxy
IMAGE=haproxy:1.9.4

DIR=`pwd -P`

docker stop $CONTAINER
docker rm $CONTAINER

docker run -d \
--net host \
--restart=always \
-v $DIR/conf:/usr/local/etc/haproxy:ro \
--name $CONTAINER \
$IMAGE

docker logs -f $CONTAINER
```

Proxying a port using HAProxy

```
global
    maxconn 100

defaults
    log global
    mode tcp
    retries 2
    timeout client 30m
    timeout connect 4s
    timeout server 30m
    timeout check 5s

listen stats
    mode http
    bind *:7000
    stats enable
    stats uri /

listen postgres
    bind *:5001
    server postgres dbhost:5432 maxconn 100 check port 5432
```

Reference

Reference	URL
HAProxy Configuration	https://cbonte.github.io/haproxy-dconv/1.9/configuration.html
Intro to HAProxy	https://www.digitalocean.com/community/tutorials/an-introduction-to-haproxy-and-load-balancing-concepts