



5

Инструкция по эксплуатации

1. Обновите пакеты репозитория в ваших машин:

```
sudo apt update & sudo apt upgrade
```

2. Импортировать файл со скриптом на вашу виртуалку, можете просто скопировать содержимое, создать файл .py и вставить туда, не забудьте сохранить

3. Запустите скрипт от рута:

```
sudo python3 file.py
```

4. Вот сам скрипт, который все сделает за вас:

```
#!/bin/python3
import os
import subprocess
import time
import re
import io
import sys
#####НЕОБХОДИМЫЕ ЗНАЧЕНИЯ#####
user = subprocess.getoutput("whoami")
ip = subprocess.getoutput("hostname -I")
#####
os.system("sudo apt install -y nginx haproxy")
##НАСТРОЙКА NGINX#####
os.system("sudo touch /etc/nginx/sites-enabled/default2")
os.system("sudo rm -rf /var/www/html")
os.system("clear")
name_dir = input("Введи название директории в которой будет храниться index.html твоего пер-
вого 'MEGA' сайта\n")
os.system("sudo mkdir /var/www/"+name_dir)
```

```

os.system("sudo touch /var/www/"+name_dir+"/index.html")
file1 = '/var/www/'+name_dir+'/index.html'
with open(file1 , "w") as f:
    f.write("<h1>SITE 1 </h1>")

while True:
    name_dir2 = input("Введи название директории в которой будет храниться index.html твое
го второго '""MEGA'"" сайта\n")
    if name_dir != name_dir2:
        os.system("sudo mkdir /var/www/"+name_dir2)
        os.system("sudo touch /var/www/"+name_dir2+"/index.html")
        break
    else:
        print("Уже существует")

file2 = '/var/www/'+name_dir2+'/index.html'
with open(file2 , "w") as f:
    f.write("<h1>SITE 2 </h1>")

file3 = '/etc/nginx/sites-enabled/default'
with open(file3 , "w") as f:
    f.write("server {\n"
    "\tlisten 81;\n"
    "\tserver_name _;\n"
    "\troot /var/www/"+name_dir+";\n"
    "\tindex index.html;\n"
    "\tlocation / {\n"
    "\t\t\ttry_files $uri $uri/ =404;\n"
    "\t}\n"
    "}")

file4 = '/etc/nginx/sites-enabled/default2'
with open(file4 , "w") as f:
    f.write("server {\n"
    "\tlisten 82;\n"
    "\tserver_name _;\n"
    "\troot /var/www/"+name_dir2+";\n"
    "\tindex index.html;\n"
    "\tlocation / {\n"
    "\t\t\ttry_files $uri $uri/ =404;\n"
    "\t}\n"
    "}")

#####НАСТРОЙКА HAProxy#####
os.system("clear")
username = input("Введите имя пользователя для мониторинга haproxy (например username): ")
password = input("Введите пароль для мониторинга haproxy (например password): ")
os.system("clear")
site1 = input("Введите название 1 сайта для его мониторинга в haproxy (например site1): ")
site2 = input("Введите название 2 сайта для его мониторинга в haproxy (например site2): ")
file8 = '/etc/haproxy/haproxy.cfg'
with open(file8 , "w") as f:
    f.write("global\n"
    "\tlog /dev/log    local0\n"

```

```
"\tlog /dev/log local1 notice\n"
"\tchroot /var/lib/haproxy\n"
"\tstats socket /run/haproxy/admin.sock mode 660 level admin expose-fd listeners\n"

"\tstats timeout 30s\n"
"\tuser haproxy\n"
"\tgroup haproxy\n"
"\tdaemon\n"

"defaults\n"
    "\tlog      global\n"
    "\tmode     http\n"
    "\toption   httplog\n"
    "\toption   dontlognull\n"
    "\ttimeout connect 5000\n"
    "\ttimeout client 50000\n"
    "\ttimeout server 50000\n"
    "\terrorfile 400 /etc/haproxy/errors/400.http\n"
    "\terrorfile 403 /etc/haproxy/errors/403.http\n"
    "\terrorfile 408 /etc/haproxy/errors/408.http\n"
    "\terrorfile 500 /etc/haproxy/errors/500.http\n"
    "\terrorfile 502 /etc/haproxy/errors/502.http\n"
    "\terrorfile 503 /etc/haproxy/errors/503.http\n"
    "\terrorfile 504 /etc/haproxy/errors/504.http\n"

"# HAproxy for web servers\n"
"frontend http_front\n"
    "bind *:80\n"
    "stats uri /haproxy?stats\n"
    "acl url_blog path_beg /blog\n"
    "default_backend backendnodes\n"

"\n"
"backend backendnodes\n"
    "\t\tbalance roundrobin\n"
    "\t\toption forwardfor\n"
    "\t\thttp-request set-header X-Forwarded-Port %[dst_port]\n"
    "\t\thttp-request add-header X-Forwarded-Proto https if { ssl_fc }\n"
    "\t\tserver "+site1+"\t"+ip.rstrip()+":81 check\n"
    "\t\tserver "+site2+"\t"+ip.rstrip()+":82 check\n"

"\n"
"listen stats\n"
    "\t\tbind :5000\n"
    "\t\tstats enable\n"
    "\t\tstats uri /\n"
    "\t\tstats hide-version\n"
    "\t\tstats auth "+username+": "+password)

os.system("sudo systemctl restart haproxy nginx")
os.system("clear")
print ("Перезапуск NGINX и HAPROXY")
time.sleep(3)
os.system("clear")
print("1. Директории с файлами сайтов index.html хранятся в /var/www/[название ваших сайтов]\n"

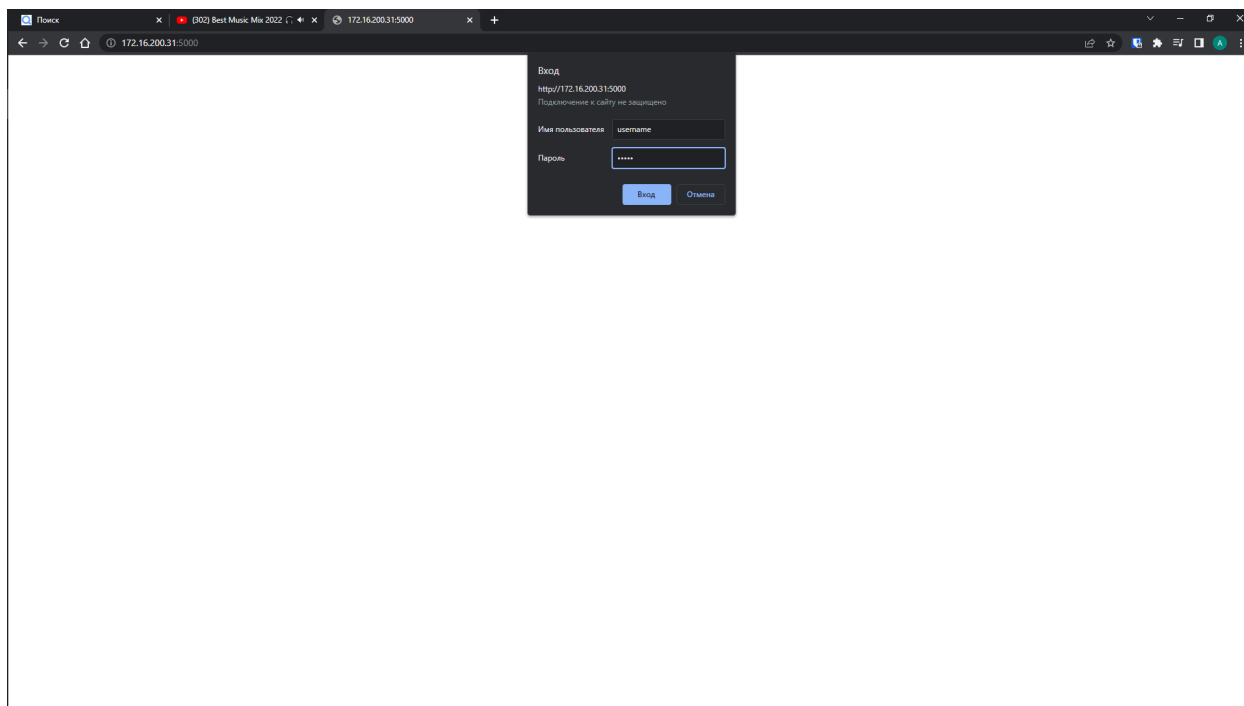
"2. Настройки вашей forward и revers зон хранятся в /etc/bind/[название вашей зоны]
```

```
forward .revers\n"
```

"3. Настройка haproxy хранится в /etc/haproxy/haproxy.cfg")

5. Проверка работы haproxy:

- После выполнения скрипта вам нужно зайти в браузер и вписать туда адрес вашей виртуалки, нажмите несколько раз на обновление страницы, должно поочередно открывать первый и второй ваш сайт, в этом собственно и есть смысл haproxy - распределение нагрузки на ваш сайт. Иными словами haproxy поочередно переводит вас на созданные виртуальные хосты в nginx, таким образом в 2 раза уменьшается количество запросов на один виртуальный хост (n виртуальных хостов - в n раз ниже нагрузка на один виртуальный хост)
- Чтобы перейти на мониторинг haproxy впишите адрес вашей виртуалки и 5000 порт, после этого запросит данные пользователя, которого вы создавали в ходе выполнения скрипта (Введите имя пользователя для мониторинга haproxy.....):



- После этого откроется панель мониторинга вашего haproxy:

Почк

(302) Best Music Mix 2022

Statistics Report for HAProxy

172.16.200.31:5000

HAProxy

Statistics Report for pid 104945

General process information

active UP

active UP, going down

active DOWN, going up

active or backup DOWN

active or backup DOWN for maintenance (MAINT)

active or backup SOFT STOPPED for maintenance

Note: "NOLOADDRAIN" = UP with load-balancing disabled.

backup UP

backup UP, going down

backup DOWN, going up

not checked

Display option:

Scope

Scope DOWN server

Status code

JSU error

External resources:

HAProxy 2.8

HAProxy 2.8.2

Online manual

HAProxy		Queue		Session rate		Sessions		Bytes		Denied		Errors		Warnings		Status		LastChk		Weight		Server								
Frontend	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LnTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Ret	Redis	Status	LastChk	Weight	Act	Bok	Chk	Den	Downtime	Throttle
Frontend	0	2	-	0	2	-	0	2	407	2			3 494	1 892	0	0	0	1				OPEN								

Backendnodes		Queue		Session rate		Sessions		Bytes		Denied		Errors		Warnings		Status		LastChk		Weight		Server								
Backend	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LnTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Ret	Redis	Status	LastChk	Weight	Act	Bok	Chk	Den	Downtime	Throttle
site1	0	0	-	0	2	-	0	1	-	2	3	12m	4 420	610	0	0	0	0	0	0	0	10m45s UP	1	1	-	-	0	0	0	-
site2	0	0	-	0	1	-	0	1	-	4	4	11m55s	2 074	855	0	0	0	0	0	0	0	10m45s UP	1	1	-	-	0	0	0	-
Backend	0	0	-	0	3	-	0	1	40	2	7	11m55s	3 494	1 871	0	0	0	0	0	0	0	10m45s UP	2	2	0	-	0	0	0	-

stats		Queue		Session rate		Sessions		Bytes		Denied		Errors		Warnings		Status		LastChk		Weight		Server								
Frontend	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LnTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Ret	Redis	Status	LastChk	Weight	Act	Bok	Chk	Den	Downtime	Throttle
Frontend	1	2	-	2	3	-	2	3	407	10			4 044	126 824	0	0	3	0	0	0	0	OPEN								
Backend	0	0	-	0	0	-	0	0	40	2	0	0	4 044	126 824	0	0	0	0	0	0	0	10m45s UP	0	0	0	0	0	0	0	-

172.16.200.31:5000/#backendnodes/site2

- В табличке backendnodes будет отображаться названия ваших сайтов которые вы вводили во время выполнения скрипта (Введите название 1 сайта для его мониторинга в haproxy....) и информация о них (количество трафика которое обрабатывает ваш виртуальный хост, его состояние (up/down))