Истомин Сергей, группа 4832, урок 3

Задание:  
1) запустить контейнер с БД, отличной от mariaDB, используя инструкции на сайте: <https://hub.docker.com/>  
2) добавить в контейнер hostname такой же, как hostname системы через переменную  
3) заполнить БД данными через консоль  
4) запустить phpmyadmin (в контейнере) и через веб проверить, что все введенные данные доступны

1 exit

2 ps aux | grep mc

3 iptables -A

4 iptables --help

5 iptables -L

6 iptables -F

7 iptables -L

8 iptables -A INPUT -p tcp --dport 22 -j ACCEPT

9 iptables -A INPUT -p tcp --dport 80 -j ACCEPT

10 iptables -A INPUT -j DROP

11 iptables-restore < /etc/iptables/rules.v4

12 iptables-restore > /etc/iptables/rules.v4

13 cd /etc/iptables

14 cd ..

15 ll

16 cd etc

17 ll

18 mkdir iptables

19 cd iptables/

20 iptables-restore > /etc/iptables/rules.v4

21 iptables -L

22 iptables -A OUTPUT -m state --stete ESTABLISHED,RELATED -j ACCEPT

23 mc

24 ps aux | grep mc

25 kill -3635

26 ps aux | grep mc

27 kill -3735

28 kill -9

29 ps aux | grep mc

30 ps -aux | grep mc

31 kill -9 3735

32 ps -aux | grep mc

33 kill -9 3837

34 ps -aux | grep mc

35 iptables -L

36 m

37 mc

38 apt install nginx -y

39 ngnix -t

40 sudo ngnix -t

41 systemctl statUS NGINX

42 systemctl status nginx

43 cd /etc/nginx

44 ll

45 nano nginx.conf

46 apt install apache2 -y

47 apache2 -t

48 sudo ngnix -t

49 ngnix -t

50 systemctl status apache2

51 nano /etc/apache2/apache2.conf

52 nano /etc/apache2/sites-enabled/000-default.conf

53 sudo sustemctl relod apache2

54 sudosustemctl relod apache2

55 systemctl reload apache2

56 systemctl start apache2

57 nginx -t

58 apache2 -t

59 nano /etc/apache2/apache2.conf

60 apache2 -t

61 apache2 -T

62 mkdir testfolder

63 rm testfolder/

64 rm -r testfolder/

65 ps aux

66 ls

67 ls /var/run/netns

68 ls /var/run

69 touch /var/run/netns/testgg

70 ps aux

71 mount --bind /proc/4143/ns/net /var/run/netns/testgg

72 nsenter --net=/var/run/netns/testgg

73 history

74 ps aux

75 unshare --net --pid --fork --mount-proc

76 ps aux

77 nsenter --target 4053 --net

78 ip a

79 ps -aux

80 nsenter --target 4053

81 mkdir testfolder

82 mkdir testfolder/bin

83 ldd /bin/bash

84 cp /bin/bash testfolder/bin

85 mkdir testfolder/lib

86 mkdir testfolder/lib64

87 cp /lib/x86\_64-linux-gnu/libtinfo.so.6 testfolder/lib

88 cp /lib/x86\_64-linux-gnu/libc.so.6 testfolder/lib

89 cp /lib64/ld-linux-x86-64.so.2 testfolder/lib64

90 chroot testfolder

91 mkdir testfolder/bin/ls

92 cp /usr/bin/ls testfolder/bin/ls

93 chroot testfolder

94 ldd ls

95 ldd /ls

96 ldd /ls/

97 ldd /usr/bin/ls

98 cp /lib/x86\_64-linux-gnu/libselinux.so.1 testfolder/lib

99 cp /lib/x86\_64-linux-gnu/libc.so.6 testfolder/lib

100 cp /lib/x86\_64-linux-gnu/libpcre2-8.so.0 testfolder/lib

101 cp /lib64/ld-linux-x86-64.so.2 testfolder/lib64

102 chroot testfolder

103 cd testfolder/

104 cd ..

105 rm -r testfolder/bin/ls

106 cp /usr/bin/ls testfolder/bin/ls

107 chroot testfolder/

108 lsns

109 man unshare

110 unshare -n

111 apt update

112 cd /

113 apt install cgroup-tools -u

114 apt update

115 apt-get install lxc debootstrap bridge-utils lxc-templates -y

116 apt update

117 find /usr-name lxc-veth.conf

118 find /usr -name lxc-veth.conf

119 lxccreate -n testcontainer

120 lxc-create -n testcontainer

121 lxc-create -n testcontainer -t ubuntu -f /usr/share/doc/liblxc-common/examples/lxc-veth.conf

122 lxc-ls -f

123 lxc start testcontainer

124 lxc-ls -f

125 lxc start testcontainer

126 lxc-start -n testcontainer

127 cd /var/run/lxc

128 ls

129 cd //

130 cd /var/lib/lxc

131 ll

132 cd testcontainer/

133 nano confif

134 ll

135 nano config

136 lxc-start -n testcontainer

137 lxc-ls -f

138 lxc-attach -n testcontainer

139 exit

140 lxc-ls -f

141 cd sys/fs/cgroup

142 cd /var/lxc/testcontainer

143 cd sys/fs/cgroup

144 lxc-ls -f

145 ls

146 lxc-stop -n testcontainer

147 lxc-ls -f

148 cd sys/fs/cgroup

149 cd /var/lib/lxc/testcontainer

150 cd sys/fs/cgroup

151 ls

152 cd /sys/fs/cgroup

153 ls

154 lxc-start -n testcontainer

155 ls

156 cd lxc.payload.testcontainer

157 ls

158 nano memory.min

159 cat memory.min

160 nano memory.max

161 cat memory.max

162 lxc-stop -n testcontainer

163 lxc-ls -f

164 lxc-start -n testcontainer

165 lxc-attach -n testcontainer

166 lxc-ls -f

167 lxc-attach -n testcontainer

168 cd /var/lib/lxc

169 cd testcontainer/

170 nano config

171 lxc-attach -n testcontainer

172 nano config

173 ls

174 lxc-attach -n testcontainer

175 nano config

176 lxc-attach -n testcontainer

177 nano config

178 lxc-stop -n testcontainer

179 lxc-ls -f

180 nano config

181 ls

182 touch logfile

183 ls

184 cat logfile

185 lxc-ls -f

186 lxc-start -n testcontainer

187 lxc-attach -n testcontainer

188 cd /sys/fs/cgroup

189 ls

190 lxc-attach -n testcontainer

191 cd /var/lib/lxc/testcontainer

192 nano config

193 lxc-start -n testcontainer

194 lxc-attach -n testcontainer

195 cd /

196 lxc-create -n container2 -t ubuntu

197 lxc-ls -f

198 cd /var/lib/lxc/

199 ls

200 cd /container2

201 cd container2

202 nano config

203 lxc-start -n container2

204 lxc-ls -f

205 nano config

206 lxc-stop -n container2

207 lxc-ls -f

208 lxc-start -n container2

209 lxc-ls -f

210 lxc-attach -n container2

211 lxc-ls -f

212 lxc-attach -n testcontainer

213 history

214 cd

215 lxc-create -n container3 -t ubuntu

216 lxc-ls -f

217 lxc-execute -n container3 /bin/bash/

218 lxc-execute -n container3 /bin/bash

219 lxc-ls -f

220 history

221 docker info

222 docker ps -a

223 docker system df

224 docker system prune -af

225 docker system df

226 docker volume ls -f dangling=true

227 docker system df

228 docker volume rm $(docker volume ls -f dangling=true -q)

229 docker system df

230 mkdir testcont

231 ls

232 docker run -it -h newcont --name ist -v /testcont:/testfolder alpine:3.16

233 cd testcont/

234 ls

235 cd

236 docker ps -a

237 docker rm ist

238 docker ps -a

239 cd testcont/

240 echo "test2 txt:" > test.txt

241 cd

242 docker run -it -h newcont --name ist -v /testcont:/testfolder alpine:3.16

243 cd testcont/

244 echo "test2 txt:" > test2.txt

245 cat test2.txt

246 docker run -it -h newcont --name ist -v /testcont:/testfolder alpine:3.16

247 docker ps -a

248 docker rm ist

249 docker ps -a

250 echo "test2 txt:22" >> test2.txt

251 docker run -it -h newcont --name ist -v /testcont:/testfolder alpine:3.16

252 ls

253 cat test2.txt

254 docker ps -a

255 docker run ist

256 docker ps -a

257 docker start ist

258 docker ps -a

259 cat test2.txt

260 docker ps -a

261 docker stop ist

262 docker ps -a

263 docker rm ist

264 docker run -it -h newcont --name ist -v /root/testcont:/testfolder alpine:3.16

265 cat test2.txt

266 cd

267 docker ps -a

268 docker start ist

269 docker exec -it ist bin/sh

270 cat /testcont/test3.txt

271 ls

272 cd testcont/

273 ls

274 cat test3.txt

275 cd

276 cd testcont/

277 cd

278 cd /

279 mkdir task1

280 cd task1

281 echo "test" >test.txt

282 cd /

283 cd

284 pwd

285 echo "test2" > test.txt

286 docker run -it -h task1 --name Task1 -v /task1:/folder1 -v /root/test.txt:/folder1/test.txt ubuntu:22.04

287 cat test.txt

288 cd /

289 cd task1/

290 ls

291 cat test.txt

292 docker run --detach --name some-mariadb --env MARIADB\_ROOT\_PASSWORD=1 mariadb:10.10.2

293 docker ps -a

294 docker stop ist

295 docker ps -a

296 docker exec -it 329965c718d6 /bin/bash

297 cd

298 cd //

299 cd /

300 docker exec -it 329965c718d6 mysql -uroot -p

301 docker run -d --name my-phpmyadmin -p 7560:80 -e PMA\_HOST=some-mariadb --link some-mariadb:db phpmyadmin/phpmyadmin:latest

302 docker ps -a

303 docker network create gusik

304 docker run -d --name db --network gusik -e MYSQL\_ROOT\_PASSWORD=1 mysql:8.0

305 docker ps -a

306 docker run --name my-adminer -d --network gusik -p 6000:8080 adminer:4.6.1

307 docker ps -a

308 docker stop c0f31b0b6aab

309 docker ps -a

310 docker rm c0f31b0b6aab

311 docker run --name my-adminer -d --network gusik -p 6000:80 adminer:4.6.1

312 docker ps -a

313 docker stop 0aad50659328

314 docker rm 0aad50659328

315 docker run --name my-adminer -d --network gusik -p 6080:8080 adminer:4.6.1

316 docker ps -a

317 docker stop 683a1f6dce2c

318 docker rm 683a1f6dce2c

319 docker run --name my-adminer -d --network gusik -p 6080:8080 adminer:4.8.1

320 history

root@serg-VirtualBox:/#