ΑΣΦΑΛΕΙΑ ΔΙΚΤΥΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ

Εργασία εξαμήνου

Ονοματεπώνυμο : Χρονάκης Μάριος
Αριθμός Μητρώου: 713242017094

Περιεχόμενα

0. Προαπαιτούμενος ενέργειες

Πρώτον, η εργασία προϋποθέτει ότι υπάρχει ήδη μια προ εγκατεστημένη διανομή LINUX είτε στο υπάρχων σύστημα σας είτε σε κάποιο virtual machine (Virtual box, VMware κ.α.).

Δεύτερο προαπαιτούμενος είναι και το docker (φυσικά !). Για την εγκατάσταση του, ανατρέξτε στον παρακάτω σύνδεσμο :

http://docs.swarmlab.io/SwarmLab-HowTos/labs/Howtos/docker/install.adoc.html#cheat-Docker

Τρίτον, απαιτείται η δημιουργία ενός swarm lab cluster για την υλοποίηση κάθε ενότητας της εργασίας και γη αυτό το λόγο είναι αναγκαίο επίσης να ανατρέξετε στον ακόλουθο σύνδεσμο:

http://docs.swarmlab.io/SwarmLab-HowTos/labs/sec/sec.adoc.html#_install_swarmlab_sec_home_pc

Με την προϋπόθεση ότι όλα τα παραπάνω υπάρχουν στο σύστημα μας, πλέον είμαστε σε θέση να ανταποκριθούμε στις ενότητες της εργασίας, καθώς έχουμε τη δυνατότητα να «σηκώσουμε τα δικά μας μηχανάκια» (με την ύπαρξη του swarm lab cluster).

1. Υλοποίηση συστήματος για την προσομοίωση DOS/DDOS Attack

1.0 Εισαγωγή

Στην πρώτη ενότητα της εργασίας θα προσομοιώσουμε τις επιθέσεις DOS (Denial Of Service) και DDOS (Distributed Denial Of Service) σ ένα worker του σμήνους (swarm).

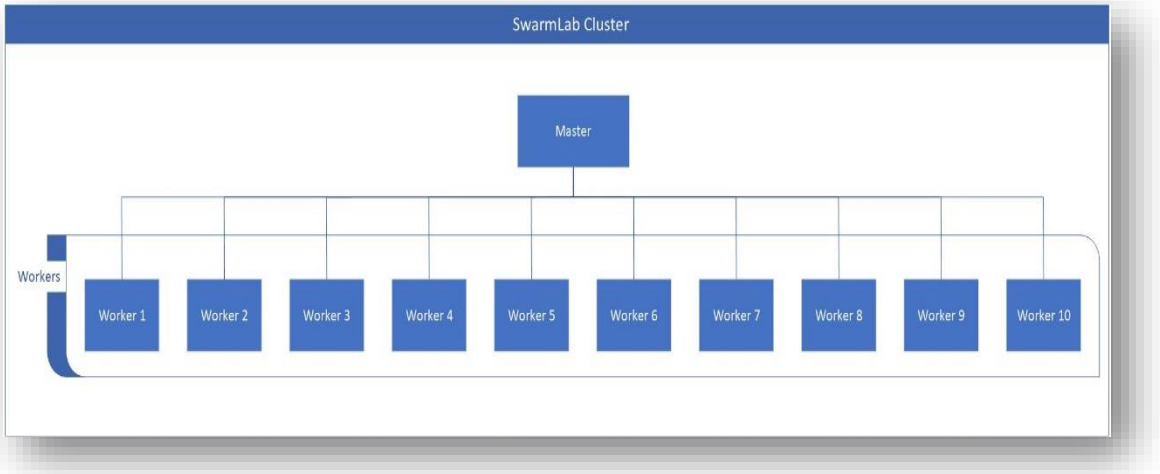
DOS attacks ή αλλιώς **Επιθέσεις άρνησης εξυπηρέτησης** ονομάζονται γενικά οι επιθέσεις εναντίον ενός υπολογιστή, ή μιας υπηρεσίας που παρέχεται, οι οποίες έχουν ως σκοπό να καταστήσουν τον υπολογιστή ή την υπηρεσία ανίκανη να δεχτεί άλλες συνδέσεις και έτσι να μην μπορεί να εξυπηρετήσει άλλους πιθανούς πελάτες. **Τα DDOS Attacks** ή αλλιώς **Κατανεμημένες Επιθέσεις άρνησης εξυπηρέτησης** δεν αποτελούν τίποτα άλλο πάρα μόνο μια προέκταση της Επίθεσης άρνησης εξυπηρέτησης (DOS attack) στις οποίες τουλάχιστον ένα τερματικό σύστημα στιχοποιοί ένα άλλο.

Οι συγκεκριμένες επιθέσεις είναι ιδιαίτερες, καθώς δε το μεγαλύτερο μέρος αυτών δεν προκαλείται από ιούς και δεν είναι δυνατόν να χρησιμοποιηθούν αντιικά προγράμματα για την αποτροπή τους. Φυσικά τα αντιικά μπορούν να ανακαλύψουν ιούς που χρησιμοποιούνται για την υποστήριξη τέτοιων επιθέσεων που όμως δεν προκαλούν άμεσο πρόβλημα σε όποιον έχει τον αντίστοιχο ιό και κατ' επέκταση δεν υποψιάζουν το θύμα-έμμεσο συνεργάτη.

DOS και DDOS attacks υλοποιούνται στις υποενότητες 1. και 1. αντίστοιχα μέσω SYN flooding (επεξηγείται στην επόμενη υποενότητα) και στην ενότητα 1. λαμβάνονται υπόψη τα καταλληλά **iprules** (μέσω **iptables**) για την αποτροπή/πρόληψη τους.

1.1 Αρχιτεκτονική συστήματος

Για τις ανάγκες της ενότητας 1, δημιουργούμε ένα σμήνος από Docker τερματικά συστήματα. Στη προκείμενη περίπτωση δημιουργήθηκε ένα σμήνος από 11 τερματικά συστήματα, εξ αυτών ένας με τον ρολό του master (ή αλλιώς manager) και 10 με τον ρολό των workers. Τα συγκεκριμένα τερματικά συστήματα βρίσκονται σε κοινό ΥΠΟ δίκτυο με address 172.23.0.0/16.



Εικόνα 1

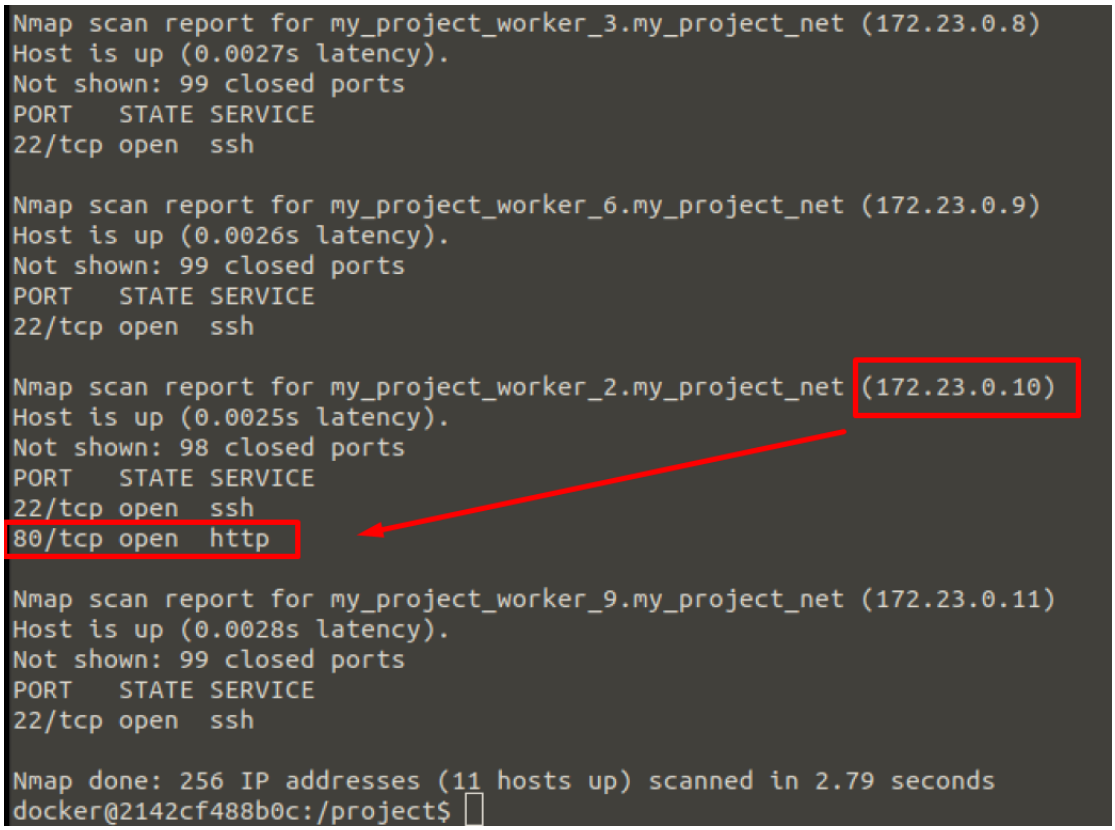
Στο συγκεκριμένο swarm cluster, ο master με address 172.23.0.2 (όπως παρουσιάζεται στην εικόνα 2 με την εντολή ifconfig) έχει πρωτεύοντα ρολό στην διεξαγωγή των επιθέσεων.

```
docker@2142cf488b0c:/project$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.23.0.2  netmask 255.255.0.0  broadcast 172.23.255.255
        ether 02:42:ac:17:00:02  txqueuelen 0  (Ethernet)
        RX packets 7380  bytes 604371 (604.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 12463  bytes 791477 (791.4 KB)
        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 26501  bytes 2004664 (2.0 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 26501  bytes 2004664 (2.0 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

docker@2142cf488b0c:/project$
```

Εικόνα 2

Πρώτου λάβει χωρά η επίθεση, είναι αναγκαίο να ερευνηθεί το ΥΠΟ δίκτυο στο οποίο ανήκει έτσι ώστε να εξακριβωθεί ο στόχος. Αυτό επιτυγχάνεται με την χρήση του εργαλείου **Nmap.** Το Nmap, συντομογραφία του Network Mapper, είναι ένα δωρεάν εργαλείο ανοιχτού κώδικα για σάρωση ευπαθειών και αναζήτηση δικτύου. Το Nmap χρησιμοποιείται να την ανακάλυψη των διαθέσιμων τερματικών συστημάτων σ ένα δίκτυο και διάφορες πληροφορίες που σχετίζονται μ αυτά όπως τις υπηρεσίες που τρέχουν, τα ports (είτε είναι σε κατάσταση OPEN ή ESTABLISHED), ακόμα και το λειτουργικό σύστημα το οποίο τρέχουν.

Εκτελώντας την εντολή **nmap -p 1-100 172.23.0.*** πραγματοποιείται σάρωση στο δίκτυο για την εύρεση ανοιχτών ports με αριθμό από 1 έως 100 σε κάθε ενεργό host. Το αποτέλεσμα της εντολής είναι εμφάνιση 11 ενεργών hosts με στοιχεία όπως IP Address, MAC Address και ανοιχτά ports (εικόνα 3).



Εικόνα 3

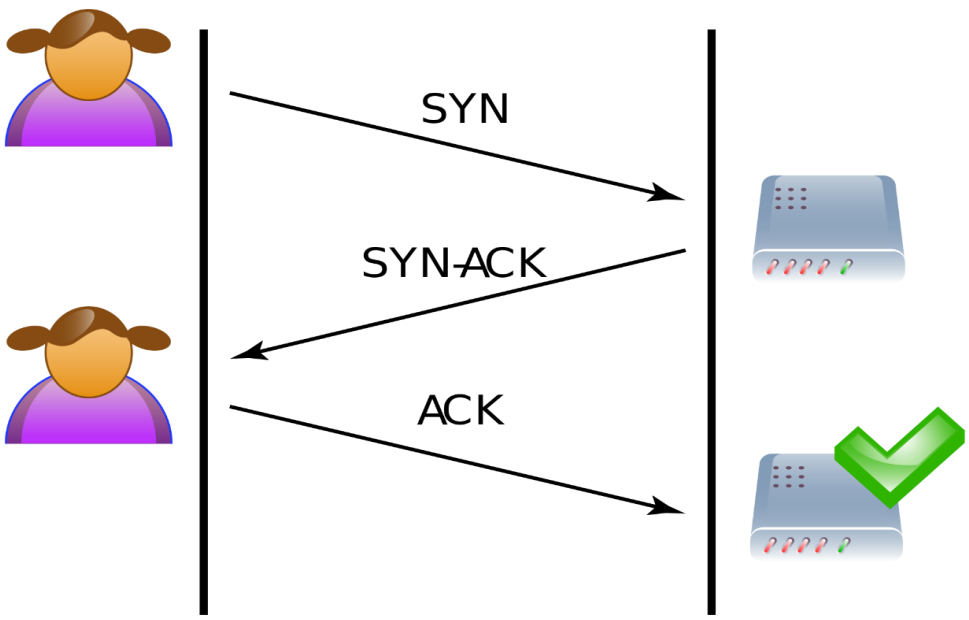
Στη ακόλουθη επίθεση στοχοποιείται ο worker 10, καθώς βρίσκεται σε κατάσταση **OPEN** το port 80 (HTTP protocol) πράγμα το οποίο καθιστά εφικτό να γίνει **πλημμύρα SYN πακέτων** (**SYN flood**).

1.2 Three-Way Handshake και SYN flood

Προτού να λάβει χωρά η επίθεση, είναι απαραίτητο να γίνει αντιληπτός ο ορός **SYN flood** που επισημάνθηκε στη προηγουμένη υποενότητα (1.1**). SYN flood** είναι ένα είδος επίθεσης άρνησης πρόσβασης (DOS - Denial of Service) κατά την οποία ο επιτιθέμενος αποστέλλει πολλαπλές αιτήσεις SYN προς το θύμα. Οι αιτήσεις SYN αποτελούν μέρος του πρωτοκόλλου επιπέδου μεταφοράς TCP.

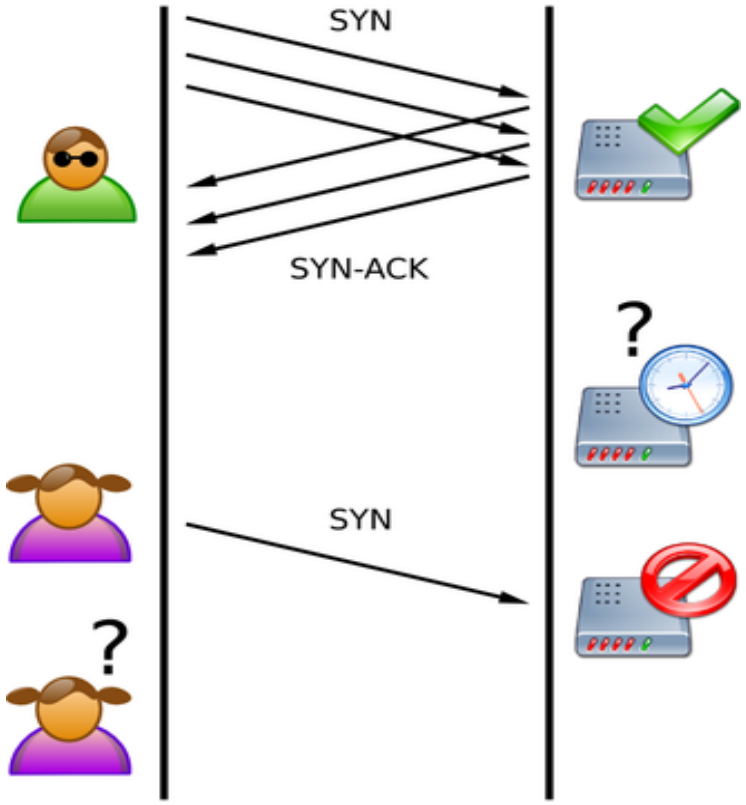
Για να γίνει κατανοητή η επίθεση είναι αναγκαίο να αντιληφθούμε την διαδικασία εγκατάστασης μια σύνδεσης μεταξύ δυο τερματικών συστημάτων μέσω TCP η οποία παραπέμπει στον ορό **three-way handshake**. Το **Three-way handshake** ή αλλιώς **τριμερής χειραψία** περιληπτικά έχει ως εξής:

- Ο πελάτης (client) ζητά την δημιουργία μίας σύνδεσης στέλνοντας έναν πακέτο TCP SYN στον διακομιστή (server). Το όνομα του πακέτου προέρχεται από την λέξη synchronize που σημαίνει συγχρονισμός.
- Ο διακομιστής απαντά στην αίτηση του πελάτη στέλνοντάς του ένα πακέτο TCP SYN-ACK, από την αγγλική λέξη acknowledge που σημαίνει αναγνώριση, αποδοχή.
- Ο πελάτης απαντά με ένα πακέτο TCP ACK δηλώνοντας ότι αποδέχεται και αυτός την σύνδεση.
- Μετά το πέρας αυτών των τριών βημάτων, η σύνδεση TCP έχει εγκαθιδρυθεί και μπορούν να αποσταλούν δεδομένα προς και από τους δύο υπολογιστές.



Εικόνα 4

Η επίθεση SYN Flood εκμεταλλεύεται το πρώτο βήμα της τριμερής χειραψίας, κατακλύζοντας με τον παραλήπτη (θύμα). Εκείνος στη προσπάθεια του να στείλει ACK πακέτα για κάθε SYN πακέτο που λαμβάνει (με αποτυχία βέβαια), καταναλώνει πόρους συστήματος και φτάνει στο σημείο να μη μπορεί να ανταποκριθεί στις προσπάθειες επικοινωνίας άλλων τερματικών συστημάτων με εκείνον.



Εικόνα 5

1.3 Διεξαγωγή επίθεσης DOS Attack

Το εργαλείο που επρόκειτο να χρησιμοποιηθεί είναι το **hping3**, το οποίο πέρα από το να στέλνει ICMP πακέτα, όπως κάνει και το κοινό **ping** προσφέρει features πάνω στα πρωτοκολλά TCP, UDP και RAW-IP.

Η εντολή για δίνεται για την πραγματοποίηση της επίθεσης είναι η ακόλουθη:

hping3 -c 10000 -d 120 -S -w 64 -p 80 --flood --rand-source 172.23.0.10

οπού,

- c 10000, ο αριθμός των πακέτων
- d 120, το μέγεθος των πακέτων
- -w 64, το TCP window size
- S, αποστολή πακέτων SYN
- p 80, η πόρτα στην οποία θα αποσταλούν τα πακέτα
- --flood, attribute που για να προκληθεί flood πακέτων
- --rand-source, για την αποστολή πακέτων με διαφορετικό source ip κάθε φορά

Σημείωση: Για την προσομοίωση του DOS attack κρίθηκε αναγκαίο να προ εγκατασταθεί στον worker 1, ένας apache2 server έτσι ώστε να γίνει η επίθεση στην πόρτα 80. Η διαδικασία εγκατάστασης έχει ως εξής :

i. ssh connection στον worker 10
ii. sudo apt update
iii. sudo apt install apache2
iv. sudo apachectl start μετά το πέρας της εγκατάστασης του apache2 server

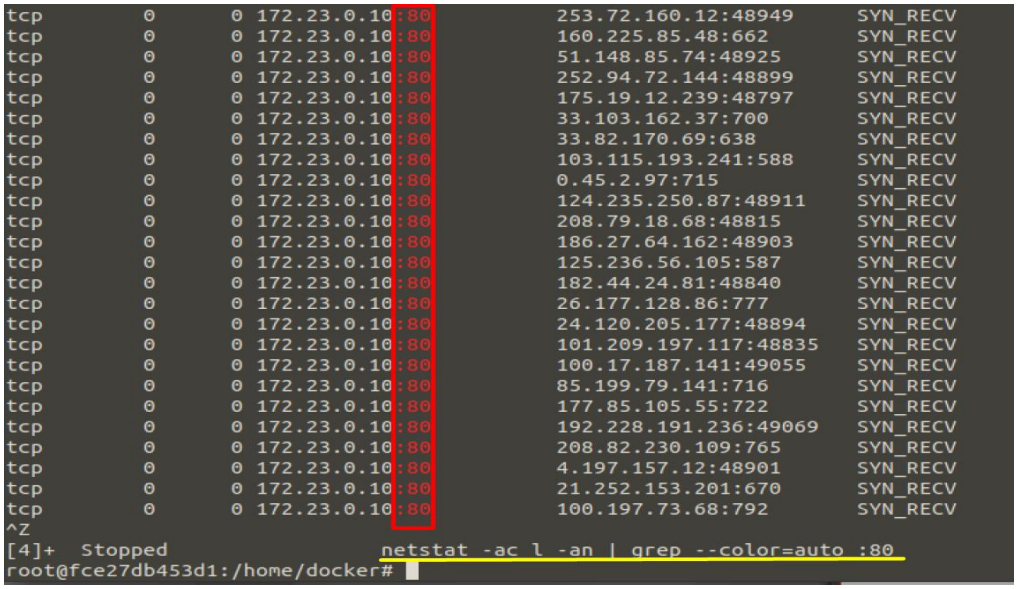
Επιπλέον, πρέπει να επισημανθεί ότι το εργαλείο hping3 είναι αναγκαίο να εγκατασταθεί στον master ως εξής:

i. sudo apt update
ii. sudo apt install hping3

1.3 Εξακρίβωση και εντοπισμός της επίθεσης

Το τερματικό σύστημα που δέχθηκε την επίθεση άρνησης υπηρεσίας (worker 10) έχει προ εγκατεστημένα στο λειτουργικό σύστημα του εργαλεία τα οποία μπορούν να χρησιμοποιηθούν για την ερευνά των αποτελεσμάτων της επίθεσης, εφόσον αυτά γίνουν αντιληπτά εξαρχής. Τα εργαλεία αυτά είναι το netstat και το tcpdump.

Πρωτίστως, το netstat είναι ένα βοηθητικό πρόγραμμα δικτύου του τερματικού που εμφανίζει συνδέσεις δικτύου για TCP, πίνακες δρομολόγησης και έναν αριθμό διεπαφών δικτύου (ελεγκτής διασύνδεσης δικτύου ή διεπαφή δικτύου καθορισμένης από λογισμικό) και στατιστικά πρωτοκόλλου δικτύου. Στη προκειμένη περίπτωσή, εκτελείται η εντολή **netstat -ac l -an | grep :80**, έτσι ώστε να εντοπιστεί ότι εισέρχεται από την πόρτα 80 (εικόνα 6).

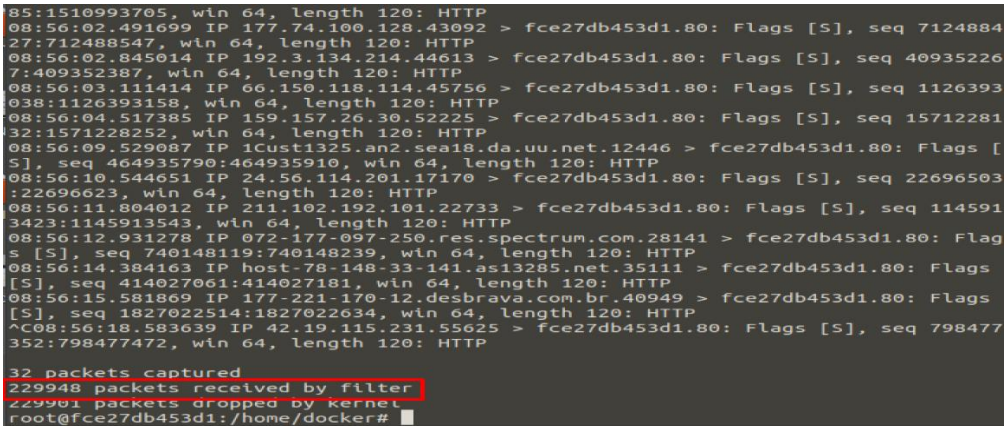


Εικόνα 6

Όπως γίνεται αντιληπτό από την εικόνα , ο apache server που τρέχει στον worker 10 λαμβάνει κατά συρροή πακέτα στην πόρτα 80. Η συγκεκριμένη κίνηση για τον συγκεκριμένο server «μικρής κλίμακας» αποτελεί ένδειξη επίθεσης και γι' αυτό ακριβώς το λόγο επόμενη κίνηση είναι η χρήση του εργαλείου tcpdump.

Το tcpdump είναι ένα εργαλείο που χρησιμοποιείται για την λεπτομερή καταγραφή πακέτων που εισέρχονται και εξέρχονται από το σύστημα. Επειδή το εργαλείο netstat εντόπισε κατακλυσμό στη πόρτα 80, άμεσο συμπέρασμα είναι εισροή SYN πακέτων λόγω TCP (three-way handshake). Επομένως για να εντοπίσουμε τα SYN πακέτα που εισέρχονται δίνουμε την εντολή tcpdump ως εξής :

tcpdump '[tcpflags] == tcp-syn'



Εικόνα 7

Τα αποτελέσματα που δίνει η εντολή (εικόνα 7) επαληθεύουν ότι ο worker 10 έχει δεχθεί DOS Attack.

1.5 Διεξαγωγή επίθεσης DDOS

Η DDOS επίθεση δεν διαφέρει σε τίποτα σε σχέση με την DOS. Η μονή διαφορά τους έγκειται στο γεγονός ότι η DDOS πραγματοποιείται με την συνύπαρξη τερματικών συστημάτων (**botnet**) τα οποία κατακλύζουν με κίνηση ένα συγκεκριμένο στόχο. Στη πραγματικότητα, attackers που επιθυμούν να διαπράξουν την συγκεκριμένη επίθεση αποστέλλουν κακόβουλο λογισμικό σε εν δύναμη θύματα, τα οποία με αν πέσουν στην παγίδα συντελούν ένα botnet. Κάθε θύμα του κακόβουλου λογισμικού αποκαλείται “bot” (εξού και η ονομασία botnet), οπού το καθένα από αυτά το «εκμεταλλεύεται» απομακρυσμένα ο attacker για να στείλει «ατέρμονα» κύματα πακέτων στόχο.

Για την προσομοίωση του DDOS attack, ο διαχειριστής του σμήνους μπορεί να πραγματοποιήσει ssh connection σε κάθε worker εκτός εκεινού που στοχοποιείται (worker 10) και να εκτελεστεί ψευδοταυτόχρονα το εργαλείο hping3 (όπως και στη προηγουμένη υποενότητα).

Διαφορετικά, δίνεται η δυνατότητα της χρήσης του εργαλείου Ansible. Το εργαλείο Ansible προσφέρει την δυνατότητα της αυτοματοποίησης ενός σμήνους σε συνδυασμό με την YAML, μια γλώσσα που χρησιμοποιείται για την συγγραφή των tasks που θα εκτελεί ένα σμήνος. Στη προκειμένη περίπτωσή, τα αρχεία **ansible.cfg** και **hosts** που βρίσκονται στο dir. **/etc/ansible** (εφόσον η ansible είναι ήδη προ εγκατεστημένη : **sudo apt install ansible**) πρέπει να ρυθμιστούν καταλληλά έτσι ώστε να αυτοματοποιηθεί η δημιουργία ενός botnet, το οποίο αποτελείται από τους workers εκτός του στόχου και θα κατευθύνεται από τον master μέσω Openssh. Στο configuration file της ansible (ansible.cfg) θέτουμε την μεταβλητή **host_key_checking** σε **False** (**host_key_checking = False),** ώστε να αποφευχθεί ο έλεγχος κλειδιού όταν γίνεται η ανταλλαγή δεδομένων μέσω **ssh.** Έπειτα στο **hosts** file πρέπει να εισαχθούν προς το τέλος του οι διευθύνσεις των containers (workers) και τα credentials που θα χρειαστούν από πλευράς ssh. Δηλαδή,

[containers]

172.23.0.3

172.23.0.4

172.23.0.5

172.23.0.6

172.23.0.7

172.23.0.8

172.23.0.9

172.23.0.10

172.23.0.11

[containers:vars]

ansible_user=docker

ansible_password=docker

Τα tasks που θα εκτελέσει το botnet για την διεξαγωγή της επίθεσης βρίσκονται στο yaml file ddos_prerequisites.yml του οποίου ο κώδικας δίνεται παρακάτω.

```
- name: installation of hping3 to each host
  hosts: containers
  remote_user: docker
  become: yes
  tasks:
  - name: apt-get update
    apt:
        update_cache: yes
  - name: install hping3
  apt:
        name: hping3
        state: latest
  - name: SYN flood via hping3
  command: hping3 -c 10000 -d 120 -S -w 64 -p 80 --flood --rand-source 172.23.0.10
```

Σημείωση: Τα συγκεκριμένα αρχεία βρίσκονται στο μονοπάτι **DDOS/Ansible/**.

1.5 Πρόληψη – Αντιμετώπιση Επίθεσης

Στην προηγουμένη υποενότητα (1.3), παρουσιάστηκαν αρκετές ενδείξεις της επίθεσης που αναλύεται στη συγκεκριμένη ενότητα. Επιπλέον, παρουσιάστηκε ότι ο worker 10 στοχοποιήθηκε γιατί η πόρτα 80 ήταν ανοιχτή, πράγμα που υποδηλώνει ότι αν ήταν κλειστή δεν θα συνέβαινε ποτέ. Αυτό το συμπέρασμα φέρνει στην επιφάνεια το πρόγραμμα firewall **iptables** του LINUX Kernel. Το **iptables** δίνει την δυνατότητα στο διαχειριστή του συστήματος να δημιουργεί κανόνες με βάση τους οποίους πρέπει να συμβαδίζουν τα εισερχόμενα και εξερχόμενα πακέτα. Οι κανόνες αυτοί δομούνται σε ομάδες ή αλλιώς **chains**. Υπάρχουν προκαθορισμένα chains για τις ανάγκες ασφάλειας του λειτουργικού συστήματος LINUX αλλά προσφέρεται η δυνατότητα στο χρήστη να δημιουργήσει τα δικά του.

Για την αντιμετωπισης επιθεσεων αρνησης υπηρεσιας DOS δημιουργηθηκε η αλυσιδα syn_flood, η οποια σχετιζεται με τα πακετα που ενδεχεται να πλημμυρησουν τον χρηστη που στοχοποιείται. Η εντολη που δινεται είναι η εξης :

sudo Iptables -N syn_flood

Οπου,

- -N, δημιουργια ένα αλυσιδας

Για να επαναληθευθουν τα τετοιου ειδους πακετα είναι αναγκαια η εκετελεση της εντολης σε admin mode :

Iptables -A syn_flood -m limit –limit 1/s –limit-burst 3 -j ACCEPT

Οπου,

- -A sflood, αλυσιδα sflood
- -m limit, module limit
- --limit 1/s, module limit το οποιο συσχετιζεται με το ρυθμο ληψης των πακετων
- -j ACCEPT, το πακετο περναει από την αλυσιδα

Οποιοδηποτε πακετο περασει από αυτή την αλυσιδα πρεπει να αποκλειστει από το συστημα και για να επιτευχθει αυτό εκτελειται η εντολη :

Iptables -A syn_flood -j DROP

Για την επαληθευση της αποτελεσματικοτητας των κανονων που δοθηκαν στο chain syn_flood πληκτρολογουμε την εντολη σε admin mode :

Iptables -nvL

Οπου,

-n, numeric format

-v, verbose output, εκτενης αναλυση κανονων

-L, εμφανιση λεπτομερειων σε λιστα

2. Υλοποίηση συστήματος για την προσομοίωση SSH Baryte Force Attack

2.0 Εισαγωγή

Το SSH (Secure Shell) είναι ένα ασφαλές δικτυακό πρωτόκολλο το οποίο επιτρέπει τη μεταφορά δεδομένων μεταξύ δύο υπολογιστών. Το SSH όχι μόνο κρυπτογραφεί τα δεδομένα που ανταλλάσσονται κατά τη συνεδρία, αλλά προσφέρει ένα ασφαλές σύστημα αναγνώρισης καθώς και άλλα χαρακτηριστικά όπως ασφαλή μεταφορά αρχείων (SSH File Transfer Protocol, SFTP), κλπ.

Το πρωτόκολλο SSH θεωρείται ασφαλές για την απομακρυσμένη σύνδεση δυο τερματικών συστημάτων, καθώς απαιτεί διαπιστευτήρια (credentials) για τη εγκαθίδρυση της. Η επαλήθευση των χρηστών γίνεται με δυο τρόπους :

- Χρήση ενός συνθηματικού το οποίο έχει οριστεί από τον server
- Χρήση ενός δημοσιου-ιδιωτικου κλειδιού βασισμένο στον γνωστό αλγόριθμο RSA.

Η χρήση ενός απλού συνθηματικού είναι πιο άμεση και ευκολότερα υλοποιήσιμη προσφέροντας ένα υποτυπώδες επίπεδο ασφάλεια εξαρτώμενο από την έκταση και την ποικιλία των χαρακτήρων του συνθηματικού. Από την άλλη η χρήση του αλγορίθμου RSA, αν και περισσότερο χρονοβόρα εγγυάται ένα μέγιστο επίπεδο ασφάλειας.

Κακόβουλοι χρήστες που επιδιώκουν να ελέγξουν απομακρυσμένα κάποιο τερματικό σύστημα μέσω του SSH, πρέπει με κάποιο τρόπο να αποκτήσουν πρόσβαση στα διαπιστευτήρια που έχουν οριστεί έτσι ώστε να το επιτύχουν. Εδώ έρχεται στην επιφάνεια η **επίθεση ωμής βίας στο SSH** ή΄αλλιως **SSH Brute Force Attack** η οποία εστιάζει σε συστήματα τα οποία χρησιμοποιούν ένα απλό συνθηματικό και με τη χρήση ενός αυτοσχέδιου λεξικού, γίνεται η προσπάθεια εύρεσης του συνθηματικού, εκτελώντας οποιονδήποτε συνδυασμό συμβολών και χαρακτήρων (με βάση το λεξικό πάντα).

Η συγκεκριμένη υποενότητα εστιάζει αποκλειστικά στο SSH, παρουσιάζοντας την επίθεση που αναφέρθηκε προηγουμένως και τρόπους αντιμετώπισης των κενών ασφάλειας του πρωτοκόλλου.

2.1 Διεξαγωγή της επίθεσης με το εργαλείο Hydra

Για την πραγματοποίηση της επίθεσης χρησιμοποιείται η διανομή KALI LINUX, στην οποία υπάρχει προ εγκατεστημένο το εργαλείο **Hydra**, το οποίο θα χρησιμοποιηθεί για την διεξαγωγή της επίθεσης. Το εργαλείο **Hydra** είναι εάν παραλληλοποιημένο login cracker που υποστηρίζει πολλά πρωτόκολλα για να επιτεθεί. Είναι πολύ γρήγορο και ευέλικτο, και οι νέες ενότητες είναι εύκολο να προστεθούν. Αυτό το εργαλείο επιτρέπει στους ερευνητές και τους συμβούλους ασφαλείας να δείξουν πόσο εύκολο θα ήταν να αποκτήσουν μη εξουσιοδοτημένη πρόσβαση σε ένα σύστημα εξ αποστάσεως.

Στη προκειμένη περίπτωσή χρησιμοποιείται για την να πραγματοποιηθεί SSH Brute Force Attack στο localhost address του συστήματος (ο τρόπος δράσης του είναι ανάλογος και σε πραγματικά συστήματα) . Πρωτίστως, πρέπει να δοθούν στο Hydra **αρχεία txt** τα οποία στην ουσία θα αποτελέσουν τα dictionaries . Για την εύρεση του **username** χρησιμοποιείται το αρχείο **top-usernames-shortlist.txt** και για την εύρεση του **password** το αρχείο **best15.txt**. Τα αρχεία αυτά βρίσκονται στον μονοπάτι **/SSH_Brute_Force_Attack/dictionaries/**.

Εκτελώντας την εντολή:

$ hydra -L top-username-shortlist.txt -P best15.txt ssh://localhost -V -t 8

οπού:

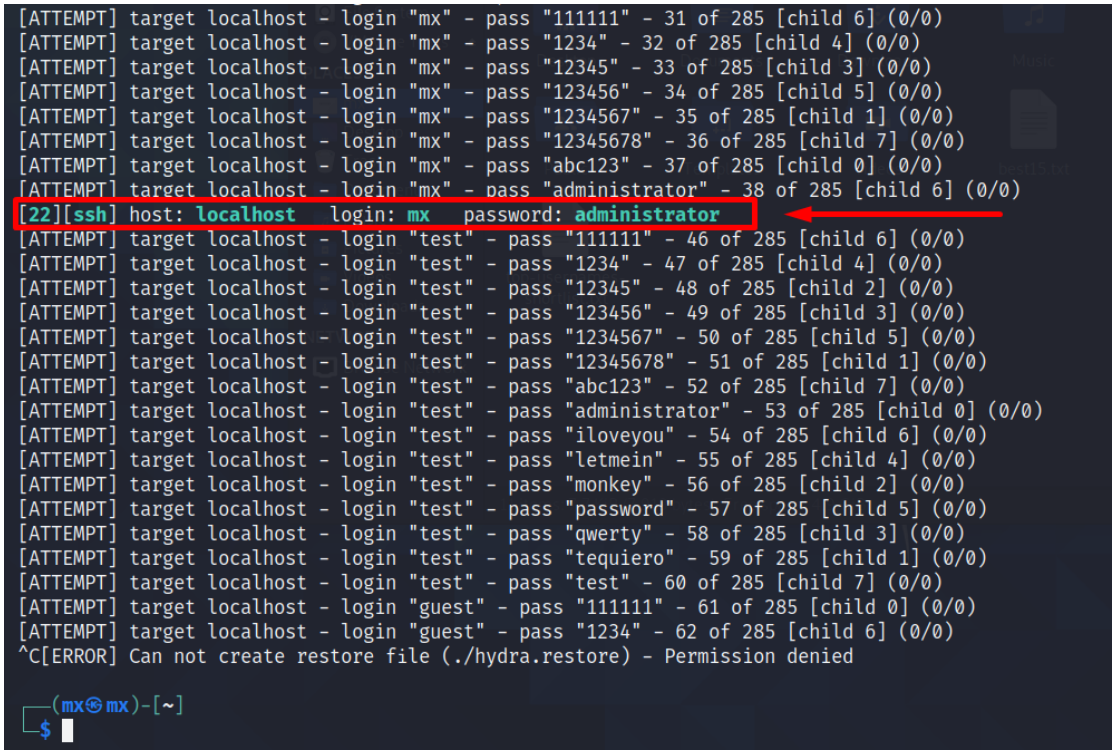
-L top-username-shortlist.txt, παράμετρος για την εισαγωγή του dictionary ` των usernames

-P best15.txt, παράμετρος για την εισαγωγή του dictionary των passwords

ssh://localhost, το πρωτόκολλο και το σύστημα που στοχοποιείται

-V, παράμετρος για την εμφάνιση των συνδυασμών

-t 8, ο αριθμός των tasks



Εικόνα 8

2.2 Αντιμετώπιση της επίθεσης με το Fail2Ban

Το Fail2Ban είναι ένα framework πρόληψης εισβολών που προστατεύει τους διακομιστές υπολογιστών από επιθέσεις ωμής βίας. Γραμμένο στη γλώσσα προγραμματισμού Python, είναι σε θέση να τρέξει σε συστήματα POSIX που έχουν μια διασύνδεση με ένα σύστημα ελέγχου πακέτων ή τείχος προστασίας εγκατεστημένο τοπικά, για παράδειγμα, iptables ή TCP Wrapper.

Σημείωση: Το Fail2Ban framework δεν προϋπάρχει στο σύστημα, επομένως είναι αναγκαίο εκτελεστούν οι εντολές sudo apt update και sudo apt install fail2ban για την ενημέρωση του συστήματος και την εγκατάσταση του εργαλείου αντίστοιχα.

Πριν προχωρήσουμε στην παραμετροποίηση του με σκοπό να προλαμβάνουμε επιθέσεις ωμής βίας στο SSH συστήνεται εκτέλεση της εντολής **sudo systemctl enable fail2ban.service**, έτσι ώστε να εκτελείται κάθε φορά που κάνουμε login στο σύστημα μας. Μεταβαίνοντας στην παραμετροποίηση του, μεταβαίνουμε στο μονοπάτι directory του με την εντολή **cd /etc/fail2ban** κι εκεί δημιουργούμε ένα αρχείο με την ονομασία **jail.local** με την εντολή **sudo touch jail.local**. Στο συγκεκριμένο αρχείο, εισάγουμε τις εγγραφές του αρχείου jail.conf με την εντολή **sudo cp jail.conf jail.local**. Τέλος κάνουμε παραμετροποίηση στο πεδίο που

απεικονίζεται στην εικόνα 8 με την εντολή **sudo nano jail.local** (οπού nano: κειμενογράφος)

Εικόνα 9

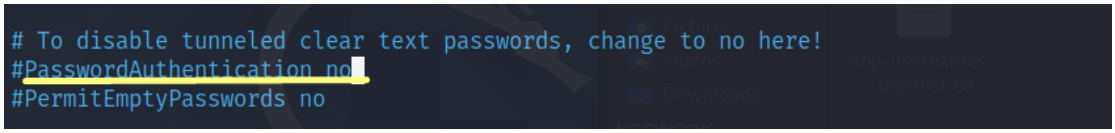
Η περιοχή που εφαρμόζονται οι τροποποιήσεις είναι η [sshd] η οποία συμβάλει στην προστασία του πρωτοκόλλου από SSH Brute Force Attacks. Κάθε παράμετρος εν συντομία :

- **enabled = true**, ενεργοποίηση ενός prison για στο οποίο το τοποθετούνται τα ip addresses των hosts που πραγματοποίησαν επίθεση.
- **port = ssh**, η πόρτα 22
- **logpath = /var/log/auth.log**, το μονοπάτι εύρεσης του αρχείου auth.log για τον εντοπισμό της δραστηριότητας.
- **backend = systemd**, ορισμός εργαλείου υποστήριξης του fail2ban framework
- **banaction = iptables**, ορισμός του firewall του LINUX KERNEL για την δημιουργία ανάλογων κανόνων
- **maxretry = 2**, ο επιτρεπτός αριθμός προσπαθειών εισαγωγής κωδικού στο SSH connection.
- **findtime = 1d**, έρευνα των attackers ανά μια μέρα
- **bantime = 3w**, αποκλεισμός των attackers για 3 εβδομάδες

Σημείωση: Το αρχείο jail.local το οποίο δημιουργούμε, έχει ακριβώς τον ίδιο ρολό με το jail.conf απλώς οι ρυθμίσεις του δευτέρου αλλάζουν σε κάθε αναβάθμιση του από το σύστημα μαζί μ εκείνες του χρήστη. Γη αυτό το λόγο, δημιουργείται και χρησιμοποιείται το jail.local το οποίο θα προτιμάται κάθε φορά από το framework για πρόληψη.

2.3 SSH connection αποκλειστικά με key

Στη συγκεκριμένη υποενότητα, θα οριστεί ο αλγόριθμος RSA που υποστηρίζεται από το πρωτόκολλο SSH για την εγκαθίδρυση συνδέσεων. Για να επιτευχθεί αυτό μεταβαίνουμε στο μονοπάτι **/etc/ssh** με την εντολή **cd /etc/ssh** και αλλάζουμε την εγγραφή **PasswordAuthentication** σε **no** στο **αρχείο sshd_config** με την εντολή **sudo nano sshd_config** (εικόνα 9).



Εικόνα 10

Πλέον ο χρήστης μπορεί να δημιουργήσει ένα ζευγάρι κλειδιών με την εντολή **ssh-keygen,** η οποία θέτει σε λειτουργία μια γεννήτρια για την παραγωγή κλειδιών μήκους 2048 bits.

3. Δημιουργία Local/Remote SSH Forwarding για την παροχή υπηρεσιών στο σμήνος

3.0 Εισαγωγή

Στην προηγουμένη ενότητα εστιάσαμε στο βασική δυνατότητα του SSH πρωτοκόλλου η οποία δεν είναι άλλη από την (ασφαλής) απομακρυσμένη σύνδεση δυο τερματικών συστημάτων. Πάρα αυτά, το πρωτόκολλο αυτό προσφέρει επιπλέον την λειτουργία tunneling μεταξύ client και server. Πιο συγκεκριμένα, ο μηχανισμός του **Port Forwarding** του πρωτοκόλλου SSH δημιουργεί ένα Tunnel μεταξύ πελάτη και εξυπηρετεί ιδιαίτερα αξιόπιστο. Υπάρχουν 2 βασικοί τρόποι υλοποίησης του συγκεκριμένου μηχανισμού : το **Local Port Forwarding** και το **Remote Port Forwarding**.

Το **Local Port Forwarding** χρησιμοποιείται για την προώθηση μιας θύρας από τον υπολογιστή-πελάτη στον υπολογιστή-διακομιστή. Βασικά, ο υπολογιστής-πελάτης SSH ακούει για συνδέσεις σε μια ρυθμισμένη θύρα, και όταν λαμβάνει μια σύνδεση, διοχετεύει τη σύνδεση με ένα διακομιστή SSH. Ο διακομιστής συνδέεται με μια θύρα προορισμού με ρύθμιση παραμέτρων, πιθανώς σε διαφορετικό υπολογιστή από το διακομιστή SSH.

Το **Remote Port Forwarding** έρχεται στην επιφάνεια όταν κάποιος μεμονωμένος χρήστης επιζητά την πρόσβαση στους πόρους ενός server ή τις υπηρεσίες απομακρυσμένα. Αυτό γίνεται πράξη, κατευθύνοντας την κίνηση από το τοπικό port προς κάποιο του απομακρυσμένου server.

3.1 Δημιουργία Local SSH Forwarding

Για τη δημιουργία ενός Local SSH Forwarding χρησιμοποιείται ένα σμήνος από 3 containers. Η address του υποδεικνύου στο οποίο ανήκουν είναι η 172.19.0.0/16. Ο master με address 172.19.0.2 θα αποτελέσει τον client και ο worker 1 με address 172.19.0.3 τον server. Ο master επιδιώκει να αποκτήσει πρόσβαση στον apache server (version 2) που τρέχει στον worker 1.

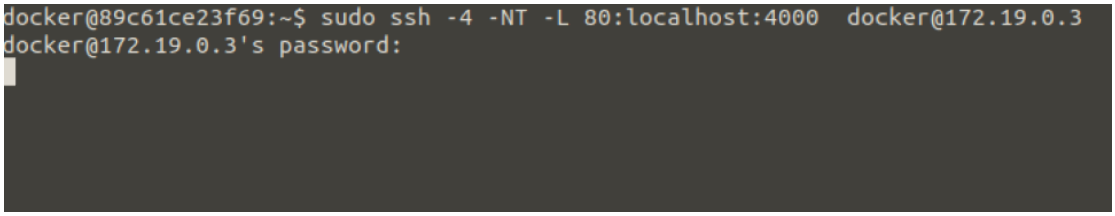
Για να το επιτύχει αυτό, εκτελεί την παρακάτω εντολή:

sudo ssh -4 -NT -L 80:localhost:4000 docker@172.19.0.3

οπού,

- -4, η επιλογή διευθύνσεων με βάση το IPv4

- -N, καμία ενέργεια μετά το ssh connection
- -T, απενεργοποίηση του pseudo-terminal allocation
- -L, Local Port Forwarding
- 80:localhost:4000, κατευθύνεται η κίνηση από την πύλη 4000 προς την πύλη 80 του localhost
- docker@172.19.0.3, ο server



Εικόνα 11

Σημείωση: Όπως γίνεται αντιληπτό στην εικόνα 10, ο master μετρά την σύνδεση του στο port 80 μέσω ssh, δεν λαμβάνει κάποια υπηρεσία γη αυτό συνεχίζει να ακούει.

3.2 Δημιουργία Remote SSH Forwarding

Για να επιτευχθεί Remote SSH Forwarding, ακολουθούμε την ιδιά διαδικασία της υποενότητα 3.1, τρέχοντας την ακόλουθη εντολή :

sudo ssh -4 -NT -R 2500:localhost:4000 docker@172.19.0.3

Όπως γίνεται αντιληπτό, έχει αντικατασταθεί η παράμετρος -L με την -R (Remote Port Forwarding) και το port 80 με 2500 οπού 2500 το port του απομακρυσμένου client (master).

4. Δημιουργία VPN για την παροχή υπηρεσιών στο σμήνος

4.0 Εισαγωγή

Ένα εικονικό ιδιωτικό δίκτυο (συνήθως αναφέρεται σαν VPN, Virtual Private Network) είναι ένα δίκτυο που χρησιμοποιεί κατά κύριο λόγο δημόσια τηλεπικοινωνιακή υποδομή, όπως το Διαδίκτυο, και δίνει τη δυνατότητα σε απομακρυσμένα γραφεία ή σε χρήστες που ταξιδεύουν να έχουν πρόσβαση σε ένα κεντρικό οργανωτικό δίκτυο.

Ένα VPN συνήθως απαιτεί από τους απομακρυσμένους χρήστες του δικτύου πιστοποίηση, και συχνά ασφαλίζει τα δεδομένα με τεχνολογίες κρυπτογράφησης για να εμποδιστεί η διάδοση των ιδιωτικών πληροφοριών σε μη εξουσιοδοτημένους τρίτους.

4.1 Δημιουργία VPN

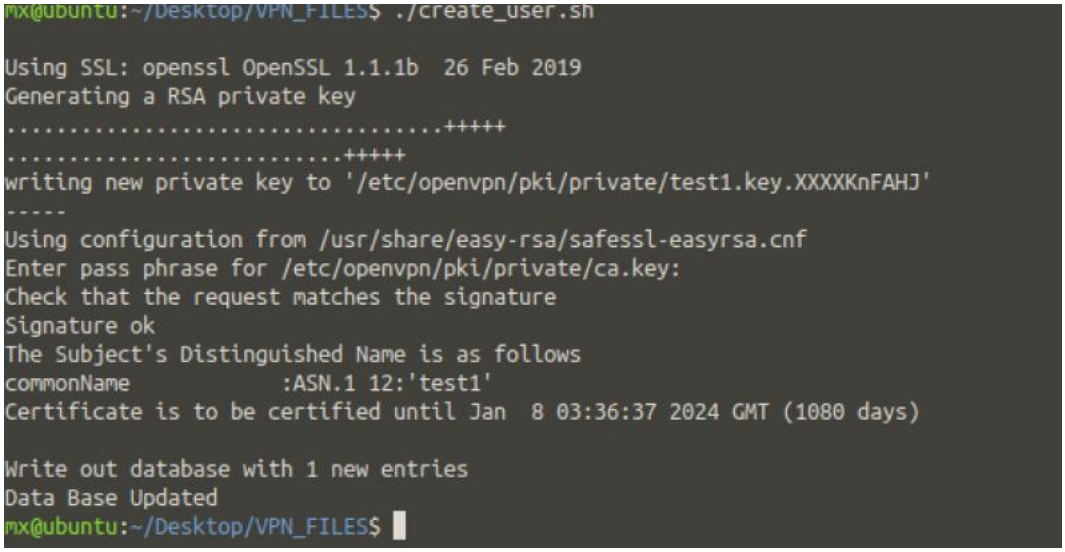
Για την δημιουργία VPN εκτελέστηκε το script με ονομασία **create_vpn.sh.** Μετά την εμφάνιση των αποτελεσμάτων εκτελέστηκε η εντολή **docker ps** έτσι ώστε να εντοπίσουμε το container το οποίο φέρει το image για τη λειτουργία του VPN server**.**

```
mx@ubuntu:~/Desktop/VPN_FILES$ docker ps
CONTAINER ID   IMAGE                                    COMMAND                  CREATED             STATUS             PORTS                    NAMES
2be1fd0f7147   registry.vlabs.uniwa.gr:5080/myownvpn    "ovpn_run"               About a minute ago  Up About a minute  0.0.0.0:1194->1194/udp   swarmlab-vpn-services
07c966e0aad7   localhost:5000/sec                       "sec_bootstrap role=…"   2 hours ago         Up 2 hours                                  my_project_worker_2
ca1104b73681   localhost:5000/sec                       "sec_bootstrap role=…"   2 hours ago         Up 2 hours                                  my_project_worker_4
a58d43fd7315   localhost:5000/sec                       "sec_bootstrap role=…"   2 hours ago         Up 2 hours                                  my_project_worker_3
f9da0b241b04   localhost:5000/sec                       "sec_bootstrap role=…"   2 hours ago         Up 2 hours                                  my_project_worker_1
89c61ce23f69   localhost:5000/sec                       "sec_bootstrap role=…"   2 hours ago         Up 2 hours         0.0.0.0:2222->22/tcp     my_project_master_1
mx@ubuntu:~/Desktop/VPN_FILES$
```

Εικόνα 12

4.2 Δημιουργία χρήστη

Το script που εκτελέστηκε για την δημιουργία χρήστη φέρει το όνομα **create_user.sh.** Τα αποτελέσματα είναι τα ακόλουθα **:**

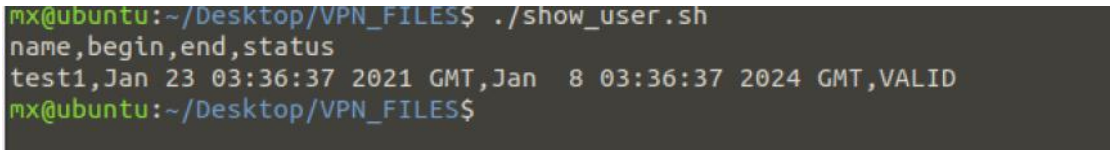


Εικόνα 13

Στο directory που εκτελέστηκε το script δημιουργήθηκε αρχείο με το όνομα **test1.ovpn**, το οποίο χρησιμοποιείται για την σύνδεση του χρήστη με VPN.

4.3 Εμφάνιση χρηστών

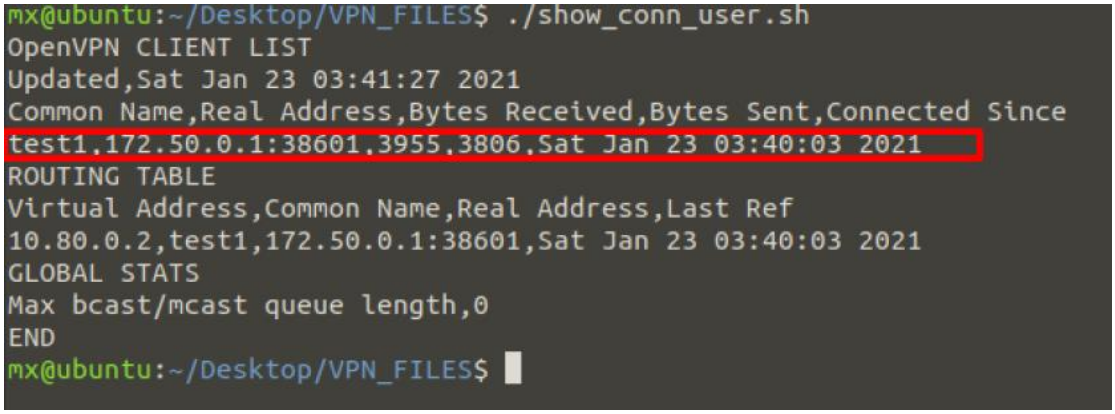
Για την εμφάνιση των χρηστών (συνδεδεμένων και μη), εκτελέστηκε το script με ονομασία **show_user.sh.** Στα αποτελέσματα που ακολουθούν στη πορεία (εικόνα 13), παρουσιάζονται πληροφορίες σχετικές με τη σύνδεση του.



Εικόνα 14

4.4 Εμφάνιση συνδεδεμένων χρηστών

Για την εμφάνιση των συνδεδεμένων χρηστών, εκτελέστηκε το script με ονομασία show_conn_user.sh. Τα αποτελέσματα είναι τα ακόλουθα:



Εικόνα 15

Σημείωση: Τα scripts που εκτελέστηκαν βρίσκονται στον κατάλογο **VPN.**

Πηγές

1. DOS/DDOS

- ddos - iptables rules to counter the most common DoS attacks? - Server Fault
- Επιθέσεις άρνησης υπηρεσιών - Βικιπαίδεια
- SYN flood - Βικιπαίδεια
- DDoS - Glossary | CSRC
- Linux Iptables Limit the number of incoming tcp connection / syn-flood attacks - nixCraft

2. SSH

- How to Gain SSH Access to Servers by Brute-Forcing Credentials « Null Byte :: WonderHowTo
- SSH - Βικιπαίδεια
- SSH Tunnel
- SSH port forwarding - Example, command, server config

3. SSH Brute Force Attack

- THC-Hydra | Penetration Testing Tools

4. VPN

- Εικονικό ιδιωτικό δίκτυο - Βικιπαίδεια
- VPN! (swarmlab.io)