

Ονοματεπώνυμο: Πυλιώτης Αθανάσιος		Όνομα PC: DESKTOP-5DLG3IF
Ομάδα: 1	Ημερομηνία: 13/05/23	

Εργαστηριακή Άσκηση 10

Τείχη προστασίας (Firewalls) και NAT

Άσκηση 1: Ένα απλό τείχος προστασίας

1.1) PC1: **kldload ipfw**

1.2) PC1: **kldstat**

1.3) PC1: **ping 127.0.0.1**

ping: sendto: Permission denied

PC1: **ping 192.168.1.2**

ping: sendto: Permission denied

Παρατηρώ πως δεν μου επιτρέπει το firewall να κάνω ping στις διευθύνσεις αυτές

1.4) PC1: **ipfw list**

65535 deny ip from any to any

1.5) PC1: **ipfw show**

Ο αύξων αριθμός του είναι 65535 άρα είναι ο τελευταίος κανόνας, 3 οι φορές που χρησιμοποιήθηκε, 252 τα bytes των ip, deny σημαίνει πως δεν δέχεται, ip σημαίνει πως δεν μας νοιάζει αν είναι IPv4 ή IPv6, from any με οποιαδήποτε διεύθυνση (και θύρα) προέλευσης, to any με οποιαδήποτε διεύθυνση (και θύρα) προορισμού.

1.6) PC1: **ipfw zero**

1.7) PC1: **ipfw add 100 allow all from any to any via lo0**

1.8) Ναι είναι επιτυχημένα τα ping από το 1.3

1.9) PC1: **ping 192.168.1.3**

ping: sendto: Permission denied

Δεν επιτρέπει να περάσει πληροφορία μέσω του LAN1.

1.10) PC1: **ipfw add allow icmp from any to any**

1.11) 00200 αύξοντα αριθμό έλαβε ο νέος κανόνας, 100 πάνω από τον προηγούμενο κανόνα που υπήρχε.

1.12) PC1: **ping 192.168.1.3**

PC2: **ping 192.168.1.2**

Ναι μπορούμε να κάνουμε και τα 2 ping.

1.13) PC1: **tracert 192.168.1.3**

Όντως δεν μπορούμε να κάνουμε tracert έτσι.

Στέλνει UDP που δεν επιτρέπονται να περάσουν.

PC1: **tracert -P icmp 192.168.1.3 (-i)**

Αν θέσουμε το πρωτόκολλο σε ICMP προφανώς και το επιτρέπει, άρα αυτή είναι η αλλαγή.

1.14) PC1: **ipfw add allow udp from me to any 33434-33625**

(φτάνει μέχρι το 33434 + ttl + probes + 1)

1.15) PC1: **ssh lab@192.168.1.3**

Ssh: connect to host 192.168.1.3 port 22: Permission denied

1.16) PC1: **ipfw add allow tcp from me to any setup**

PC1: **ipfw add allow tcp from any to any established**

Δουλεύει!

1.17) PC1: **ipfw zero → ssh lab@192.168.1.3 → lab@PC2: ls → exit**

1.18) PC1: **ipfw show →**

Ο 2 θα εκτελεστεί μία φορά, όπως ήταν αναμενόμενο επειδή το setup γίνεται μία φορά για την TCP σύνδεση και ο 1 εκτελείται 80 φορές, πλήθος των TCP πακέτων που ανταλλάχθηκε.

1.19) PC2: **ssh lab@192.168.1.2**

Όχι δεν πετυχαίνει καθώς το firewall απορρίπτει εισερχόμενα από όχι established σύνδεση

1.20) PC2: **service ftpd onestart**

1.21) PC1: **ftp lab@192.168.1.3**

Συνδεθήκαμε και καταφέραμε να κατεβάσουμε κάτι.

PC1: **ftp> get /usr/bin/znew /test**

επιτυχές

Άσκηση 2: Ένα πιο σύνθετο τείχος προστασίας

2.1) PC2: **kldload ipfw**

2.2) PC2: **ping 192.168.1.2**

Όχι δεν μπορούμε

2.3) PC2: **ipfw add allow all from any to any via lo0**

2.4) PC2: **ipfw add allow icmp from me to any icmptypes 8**

2.5) PC2: **ping 192.168.1.2**

Όχι ακόμα, δεν μπορούμε. Παρατηρώ πως τα ICMP Echo Reply στέλνονται αλλά δεν τα επιτρέπει το firewall στο PC2.

2.6) PC2: **ipfw show**

Ενεργοποιήθηκε πολλές φορές, που μας δείχνει πως τα ICMP Echo Request διέρχονται από το firewall. Ωστόσο, δεν γίνονται δεκτά τα ICMP Echo Reply και για αυτό αποτυγχάνει το ping.

2.7) PC2: **ipfw delete 200**

PC2: **ipfw add allow icmp from me to any icmptypes 8 keep-state**

Ναι μπορούμε

2.8) PC1: **ping 192.168.1.3**

Ναι μπορούμε, τρέχουν ταυτόχρονα.

2.9) Ctrl+C, περιμένουμε και συνεχίζει ακόμα κανονικά. Σταματάμε και στο PC1 και ξανα αρχίζουμε. Πλέον δεν μπορούμε να κάνουμε ping από το PC1 στο PC2.

Αυτό συμβαίνει επειδή το keep-state επέτρεπε να έρχονται πακέτα από ήδη established σύνδεση, οπότε το PC1 έστειλε επειδή υπήρχε ήδη σύνδεση, αλλά με το που διακόπηκε σταμάτησε να μπορεί να στέλνει.

2.10) PC2: **ipfw add allow icmp from any to me icmptypes 8 keep-state**

2.11) PC1: **ping 192.168.1.3**

PC2: **ipfw -d show**

Παρατηρούμε τους 4 στατικούς κανόνες που υπάρχουν (100, 200, 300, 65535) και τον δυναμικό κανόνα από τον κανόνα 300 που είναι:

```
00300 90 7560 (4s) STATE icmp 192.168.1.2 0 <=> 192.168.1.3 0 : default
```

2.12) PC2: **ipfw -d show**

Παρατηρούμε πως μετά από 4 second ο δυναμικός κανόνας σταματάει να υπάρχει.

2.13) PC2: **ipfw add allow udp from any to me 33434-33625**

PC2: **ipfw add allow icmp from me to any icmptypes 3**

Δουλεύει το traceroute προς το PC2.

2.14) PC2: **ipfw add allow udp from me to any 33434-33625**

PC2: **ipfw add allow icmp from any to me icmptypes 3**

2.15) PC1: **ipfw add allow udp from any to me**

2.16) PC2: **ipfw add allow tcp from 192.168.1.0/24 to me 22 keep-state**

2.17) PC1: **ssh lab@192.168.1.3**

Πετυχαίνει

2.18) PC2: **ipfw add allow tcp from me to any 22 keep-state**

2.19) PC1: **ipfw add allow tcp from 192.168.1.3 to me 22 setup**

2.20) PC1: **sftp lab@192.168.1.3**

Ναι μπορούμε.

2.21) PC1: **ftp lab@192.168.1.3**

Όχι δεν μπορούμε γιατί το ftp συνδέεται μέσω της θύρας 21 και όχι της 22

PC1: **ipfw add allow tcp from any to me 21 keep-state**

2.22) PC1: **ftp lab@192.168.1.3**

PC1: ftp> **cd /usr**

PC1: ftp> **ls**

Δεν μπορούμε επειδή χρησιμοποιεί τη θύρα 20 γιατί είναι σε passive mode αντί της θύρας 21

2.23) PC2: **ipfw add allow tcp from any 1024-65535 to me 1024-65535 setup keep-state**

2.24) ftp> **get /usr/bin/tee /test**

Ναι μπορούμε γιατί σε passive mode ο client στέλνει πακέτα μέσω της θύρας 21 που επιτρέπουμε.

2.25) PC2: **ipfw add allow tcp from me 20 to any setup keep-state**

PC1: **ipfw add allow tcp from any 20 to me setup**

2.26) Το FTP με το firewall δεν τα πάει πολύ καλά και θέλει πολλούς κανόνες σε ελέγχους κανόνων και δεδομένων. Χρειάζεται πολλά με τους clients και τους servers και ακόμα περισσότερο σε active mode.

2.27) PC1: **kldunload ipfw → kldstat**

PC2: **kldunload ipfw → kldstat**

Ναι είναι unloaded

Άσκηση 3: Απλό Network Address Translation

3.1) PC1(2): **hostname PC1(2) → ifconfig em0 192.168.1.2(3)/24 → route add default**

192.168.1.1

3.2) R1: **cli → router.ntua.lab# configure terminal → router.ntua.lab(config)# hostname R1 → R1(config)# interface em0 → R1(config-if)# ip address 192.0.2.2/30 → R1(config-if)# exit → R1(config)# interface em1 → R1(config-if)# ip address 192.0.2.6/30**

3.3) SRV1: **hostname SRV1 → ifconfig em0 192.0.2.5/30 → route add default 192.0.2.6**

3.4) PC2/SRV1: **service ftpd onestart**

3.5) FW1: **kldstat**

intpm.ko

smbus.ko

ipfw.ko

ipfw_nat.ko

libalias.ko

3.6) Service -e

Firewall_enable="YES" → ipfw at boot time, έχει ενεργοποιηθεί το firewall

3.7) Η λειτουργία του τείχους προστασίας είναι unknown.

3.8) FW1: **ipfw list**

12 κανόνες βλέπουμε και τελευταίος κανόνας είναι ο 65535 deny ip from any to any

3.9) FW1: **ipfw nat show config**

Όχι δεν υπάρχει κάτι

3.10) PC1: **ping 192.168.1.1**

PC1: **ping 192.0.2.1**

Όχι δεν μπορούμε να κάνουμε ping σε κανένα από αυτά τα δύο.

3.11) SRV1: **ping 192.0.2.1**

Όχι δεν μπορούμε.

3.12) FW1: ipfw nat 123 config ip 192.0.2.1 reset unreg_only

3.13) FW1: **ipfw add nat 123 ip4 from any to any**

3.14) PC1: **ping 192.168.1.1**

PC1: **ping 192.0.2.1**

Ναι μπορούμε.

3.15) R1: **tcpdump -vvv -en -i em0**

3.16) FW1: **ipfw show, ipfw zero**

3.17) PC1: **ping -c 3 192.0.2.2**

Παρατηρούμε τη διεύθυνση IP 192.0.2.1 (FW1 στο WAN1) ως διεύθυνση πηγής των ICMP Echo Request.

3.18) ICMP Echo Reply έχει διεύθυνση προορισμού την 192.0.2.1 (FW1 στο WAN1)

3.19) Ο κανόνας **nat 123 ip4 from any to any**

3.20) Ο κανόνας εφαρμόστηκε 12 φορές, 6 για τα ICMP Echo Request και 6 για τα ICMP Echo Reply. 3 από αυτά φεύγουν από το PC1 και περνάνε από το FW1, και επιστρέφουν 3 ICMP Echo Replies από το R1 και κάθε ένα μεταφράστηκε από 2 φορές όταν περνούσαν από το FW1 (μπαίνουν και βγαίνουν)

3.21) SRV1: **ping -c 1 192.0.2.1**

Ναι μπορούμε

3.22) Ο κανόνας **nat 123 ip4 from any to any**

```
root@PC:~ # ipfw list
00100 allow ip from any to any via lo0
00200 deny ip from any to 127.0.0.0/8
00300 deny ip from 127.0.0.0/8 to any
00400 deny ip from any to ::1
00500 deny ip from ::1 to any
00600 allow ipv6-icmp from :: to ff02::/16
00700 allow ipv6-icmp from fe80::/10 to fe80::/10
00800 allow ipv6-icmp from fe80::/10 to ff02::/16
00900 allow ipv6-icmp from any to any icmp6types 1
01000 allow ipv6-icmp from any to any icmp6types 2,135,136
01100 nat 123 ip4 from any to any
65535 deny ip from any to any
```

3.23) **Ναι ωθείται** καθώς προέρχεται από ιδιωτική διεύθυνση, αφού ο NAT ορίζεται σαν any to any και όλα περνούν από αυτό.

3.24) PC2: **ssh lab@192.0.2.5**

Ναι μπορούμε.

3.25) SRV1: **ssh lab@192.168.1.3**

Κατά τη προσπάθεια εμφανίζεται μήνυμα 'no route to host' συνεπώς καταλαβαίνουμε πως είναι πρόβλημα δρομολόγησης επειδή ο R1 δεν έχει default route ούτε γνωρίζει την ύπαρξη του LAN1. (ενώ ο PC2 έχει default route)

3.26) FW1: **ipfw nat 123 config if em1 reset unreg_only redirect_addr 192.168.1.3 192.0.2.1**3.27) SRV1: **ssh lab@192.0.2.1**

Ναι είναι επιτυχής, το καταλαβαίνουμε από το hostname και είμαστε στο PC2

3.28) FW1: **ipfw nat 123 config if em1 reset unreg_only redirect_addr 192.168.1.3 192.0.2.1 redirect_port tcp 192.168.1.2:22 22**3.29) SRV1: **ssh lab@192.0.2.1**

Ναι είναι επιτυχής, το καταλαβαίνουμε από το hostname και είμαστε στο PC1

3.30) SRV1: **ftp lab@192.0.2.1**

Συνδέεται στο PC2 (από το PC2 FTP Server όταν γίνεται η σύνδεση) καθώς πάει στο PC1 μόνο για το port 22.

3.31) SRV1: **ftp> ls /etc**

Ναι μπορούμε

3.32) PC1: **ftp lab@192.0.2.1**

Απαντάει το PC2

3.33) PC2: **ssh lab@192.0.2.1**

Συνδεθήκαμε από το PC1

Άσκηση 4: Τείχος προστασίας και NAT

4.1) FW1: **ipfw disable one_pass**

PC1: **ping 192.168.1.1**

SRV1: **ping 192.0.2.1**

Όχι δεν γίνεται ping

4.2) Ναι γίνονται δεκτά, αλλά υπάρχει ένας τουλάχιστον κανόνας που το απορρίπτει (χρειάζεται να περάσει από όλους τους κανόνες) που είναι ο deny all

4.3) FW1: **ipfw list → ipfw delete 1100 →**

FW1: **ipfw add 1100 allow all from any to any via em0**

4.4) Ναι το ping από το PC1 τώρα είναι επιτυχές.

4.5) PC2: **ssh lab@192.0.2.1**

Στο FW1 καθώς δεν υπάρχει πλέον ο κανόνας για τον in-kernel πίνακα NAT.

4.6) Ο κανόνας που προσθέσαμε παραπάνω και επιτρέπει όλη τη κίνηση μέσω του em0 να περνάει.

4.7) FW1: **ipfw add 3000 nat 123 all from any to any xmit em1**

4.8) FW1: **ipfw add 3001 allow all from any to any**

4.9) FW1: **ipfw add 2000 nat 123 all from any to any recv em1**

4.10) FW1: **ipfw add 2001 check-state**

4.11) PC1: **ping 192.0.2.1**

Απαντάει η ίδια διεύθυνση, το FW1 (λόγω του κανόνας 1100)

4.12) SRV1: **ping 192.0.2.1**

Το PC2 (λόγω του κανόνας 2000)

4.13) PC1: **ssh lab@192.0.2.1**

Απαντάει η ίδια διεύθυνση, το FW1 (λόγω του κανόνας 1100)

4.14) SRV1: **ssh lab@192.0.2.1**

Το PC1

4.15) SRV1: **ftp lab@192.0.2.1**

Το PC2

4.16) PC1: **ping 192.0.2.5**

Ναι μπορούμε

4.17) PC1: **ssh lab@192.0.2.5**

Ναι μπορούμε

4.18) PC1: **ftp lab@192.0.2.5**

ftp> ls

ftp> get /usr/bin/znew /test

Ναι μπορούμε να τα κάνουμε όλα αυτά

4.19) FW1: ipfw add 2999 deny all from any to any via em1

4.20) Πετυχαίνουν μόνο όσα είναι από το PC1 προς το FW1, δηλαδή όσα είναι μόνο στο LAN1 και έχουν διεπαφές.

4.21) FW1: ipfw add 2500 skipto 3000 icmp from any to any xmit em1 keep-state

4.22) Ναι μπορούμε (PING PC1 → SRV1)

4.23) FW1: ipfw add 2600 skipto 3000 tcp from any to any 22 out via em1 keep-state

4.24) Ναι μπορούμε (ssh PC1 → SRV1)

4.25) FW1: ipfw add 2100 skipto 3000 icmp from any to any in via em1 keep-state

4.26) SRV1: ping 192.0.2.1

Απαντάει το PC2

4.27) FW1: ipfw add 2200 skipto 3000 tcp from any to any 22 recv em1 keep-state

4.28) SRV1: ssh lab@192.0.2.1

Συνδέεται στο PC1

4.29) SRV1: ftp lab@192.0.2.1

Όχι δεν πετυχαίνει

4.30) FW1: ipfw add 2300 skipto 3000 tcp from any to any 21 recv em1 keep-state

FW1: ipfw add 2400 skipto 3000 tcp from any 20 to any out via em1 keep-state

Άσκηση 5: Τείχος προστασίας με γραφικό περιβάλλον διαχείρισης

Τα παρακάτω αναφέρονται στο M0n0wall

5.1) Interface > LAN > **192.168.1.1/24**

5.2) Interface > WAN > **10.0.0.1/30**

5.3) Status > System > 34% **δεσμευμένη μνήμη**, ελεύθερη μνήμη **66%**

5.4) Status > Interfaces > Βλέπουμε **4 διεπαφές**, οι διεπαφές ήταν λίγο λάθος αλλά πλέον είναι

LAN1, WAN1, Host-only και DMZ, με αυτή τη σειρά.

5.5) Interfaces > DMZ > **172.22.1.1/24**

5.6) System > General setup > Hostname: **fw**

5.7) System > General setup > Hostname: **fw1** → **save** (στο τέλος της σελίδας)

5.8) Firewall > Rules > WAN > **όχι** δεν υπάρχουν κανόνες

5.9) Interfaces > WAN >

IP address: **192.0.2.1/30**

Gateway: **192.0.2.2**

Block Private Networks → **tick**

Save

5.10) Firewall > Rules > WAN **Ναι** υπάρχει ένας κανόνας, το “Block private networks”

5.11) Services > καμία δεν φαίνεται να είναι ανοιχτή

5.12) Services > **Enable DNS forwarder** → **Save**

5.13) Services > DHCP server > LAN > **Enable** → **Range 192.168.1.2 to 192.168.1.3**

5.14) PC1: **dhclient em0**

Αποδίδονται: IP: 192.168.1.2,

Netstat -rn → Gateway: 192.168.1.1,

Cat etc/resolv.conf → DNS Server: 192.168.1.1

5.15) Σύμφωνα με τη σελίδα του, αν είναι ενεργοποιημένο το DHCP Service θα προσδώσει αυτόματα στο LAN στην IP σαν DNS Server από DHCP Clients, άρα θα χρησιμοποιήσουν το forwarding.

To forwarding χρειάζεται για να είναι αυτόματο. Πρακτικά το firewall λειτουργεί και σαν DNS server.

5.16) Diagnostics > Logs > DHCP > αναγράφει ακριβώς ποια διεύθυνση δόθηκε και που.

5.17) Diagnostics > ARP table > **5** εγγραφές παρατηρούμε.

5.18) PC1: ping 192.168.1.1

Όχι δεν μπορούμε

5.19) Diagnostics > Logs > Firewall παρατηρούμε ένα error log list και μας εμφανίζει τις 50

τελευταίες καταγραφές, που είναι όλες deny. Παρατηρούμε όλα τα πακέτα που έχουν

απορριφθεί, μαζί και τα πακέτα icmp από το ping του PC1 που απορρίφθηκαν. → **Clear log**

5.20) Diagnostics > Firewall states > 2 παρατηρούμε.

5.21) Firewall > Rules> LAN > κανένας κανόνας για το LAN.

5.22) Firewall > Rules> LAN > “Add new rule” → Action: pass, Interface: LAN, Protocol: any, Source : any, Source port range: any, Destination: any, Destination port range: any. No fragments, not Log → **Save → Apply changes**

5.23) PC1: **ping 192.168.1.1 (192.0.2.2), (172.22.1.1)**

Ναι μπορούμε και στα 3


5.24) R1(router)# **do ping 192.0.2.1**

Όχι δεν μπορώ

5.25) R1: **arp -a**

Ναι βλέπουμε, είναι η 08:00:27:e9:e7:7f MAC of firewall fw1 at WAN1

5.26) Firewall > Rules> WAN> “Add new rule” → Action: pass, Interface: WAN, Protocol: ICMP, Source: any, Source port range: any, Destination IP: WAN address, Destination port range: any. No fragments, not Log → **Save → Apply changes**

	ICMP	*	*	WAN address	*	
---	------	---	---	-------------	---	--

5.27) R1(router)# **do ping 192.0.2.1**

Ναι μπορώ

5.28) R1(router)# **do ping 192.168.1.2**

Όχι δεν μπορώ no route to host γιατί δεν υπάρχει διαδρομή στον R1 για το PC1 γράφει, αλλά στη πραγματικότητα δεν μπορώ επειδή επιτρέπω κίνηση μόνο προς το WAN address.

5.29) PC1: **ping 192.0.2.2**

Ναι μπορώ. Η NAT προστατεύει και το ιδιωτικό δίκτυο όσο ένας από τους εσωτερικούς υπολογιστές θα μπορούσε να ξεκινήσει επικοινωνία με το δημόσιο internet, χωρίς να είναι προσβάσιμο από εξωτερικό υπολογιστή, προσφέροντας ασφάλεια και ανωνυμία.

5.30) SRV1: **ifconfig em0 172.22.1.2/24**

Όχι δεν μπορούμε να κάνουμε ping στο PC1 γιατί δεν είναι στο ίδιο δίκτυο και δεν έχει οριστεί default gateway άρα έχει no route to host.

5.31) SRV1: **route add default 172.22.1.1**

5.32) PC1: **ping 172.22.1.2**

Ναι μπορούμε

5.33) SRV1: **ping 172.22.1.1**

Όχι δεν μπορούμε καθώς δεν υπάρχει κατάλληλος κανόνας στο firewall που να το επιτρέπει. και δεν επιτρέπει το ICMP Echo Request

5.34) SRV1: **ping PC1, R1**

Όχι γιατί δεν έχουμε κατάλληλο κανόνα που να μπορεί να στείλει πακέτο από τη διεπαφή στο DMZ, και δεν επιτρέπει το ICMP Echo Request

5.35) Firewall > Rules> DMZ> “Add new rule” → Action: pass, Interface: DMZ, Protocol: any, Source: DMZ subnet, Source port range: any, Destination IP: !LAN subnet, Destination port range: any. No fragments, not Log → **Save → Apply changes**

5.36) SRV1: **ping 172.22.1.1**

Ναι μπορούμε

5.37) SRV1: **ping 192.0.2.1**

Ναι μπορούμε

5.38) R1(config)# **do ping 172.22.1.2**

Όχι, no route to host, άρα δεν ξέρει διαδρομή για να πάει σε αυτό.

5.39) SRV1: **ping 192.0.2.2**

Ναι μπορούμε γιατί υπάρχει default gateway που το στέλνει Στο FW1 και εκείνο ξέρει πως να πάει στο WAN1. Επίσης υπάρχει κανόνας που επιτρέπει τη κίνηση μέσω της διεπαφής DMZ προς όλα τα δίκτυα εκτός του LAN1. Και μετά προωθεί τη κίνηση μεταφρασμένη από το DMZ στο R1.

5.40) PC2: **dhclient em0**

Η διεύθυνση IP που δόθηκε είναι η 192.168.1.3, Η default gateway 192.168.1.1 και η διεύθυνση εξυπηρέτησης DNS είναι 192.168.1.1

5.41) Firewall > Rules> LAN> “Add new rule” → Action: block, Interface: LAN, Protocol: any, Source IP: single-host 192.168.1.3, Source port range: any, Destination IP: single-host 172.22.1.2, Destination port range: any. No fragments, not Log → **Save → Apply changes**

5.42) Πρέπει να τοποθετηθεί πριν αλλιώς θα επιτρέπεται όλη η κίνηση από το LAN1. Άρα το επιλέγουμε και πατάμε move selected rules before this one στο πάνω και πατάμε apply changes.

5.43) PC2: **ping 172.22.1.2**

Όχι δεν μπορώ

5.44) PC2: **ping 172.22.1.1**

Ναι μπορώ, επειδή η εγγραφή που μόλις βάλαμε μπλοκάρει πακέτα μόνο από το PC2 στο SRV1 και επιτρέπει όλα τα άλλα.

Άσκηση 6: Τείχος προστασίας και προχωρημένο NAT

6.1) R1(config)# **ip route 203.0.118.0/24 192.0.2.1**

6.2) Firewall > NAT > outbound > Enable advanced outbound NAT → **Save**

6.3) Firewall > NAT > outbound > Interface: WAN, Source: 192.168.1.2/32, Destination: any,
Target: 203.0.118.14 → **Save** → **Apply changes**

6.4) Firewall > NAT > outbound > Interface: WAN, Source: 192.168.1.3/32, Destination: any,
Target: 203.0.118.15 → **Save** → **Apply changes**

6.5) R1: **tcpdump -vvv -en -i em0**

6.6) PC1: **ping 192.0.2.2**

Ναι μπορούμε, IP Source: 203.0.118.14

PC2: **ping 192.0.2.2**

Ναι μπορούμε, IP source: 203.0.118.15

6.7) R1: **ping 203.0.118.14**

Αποτυγχάνει, Η μετάφραση NAT είναι outbound και το FW1 δεν έχει εγγραφή για τη προώθηση αυτών των πακέτων προς το PC1, φαίνεται και στα firewall error logs.

6.8) Firewall > NAT > server NAT > 203.0.118.18 → **Save** → **Apply changes**

6.9) Firewall > NAT > inbound > Interface: WAN, External address: 203.0.118.18(), Protocol: TCP, external port range: SSH, NAT IP: 172.22.1.2 local port: SSH, Auto-add a firewall rule to permit traffic through this NAT rule: enabled → **Save** → **Apply changes**

6.10) Firewall > Rules > WAN

Τοποθετείται κανόνας που επιτρέπει TCP κίνηση προς το 172.22.1.2 μέσω της θύρας 22 με περιγραφή NAT, ώστε να μπορεί να γίνει TCP σύνδεση με το SRV1. Προστέθηκε λόγω του auto-add a firewall rule που επιλέξαμε.

6.11) R1: **ssh lab@203.0.118.18**

Ναι μπορούμε, και συνδεόμαστε στο Hostname → SRV1, άρα στον SRV1

6.12) R1: **ping 203.0.118.18**

Δεν πετυχαίνει επειδή υπάρχει κανόνας που επιτρέπει μονάχα τη TCP σύνδεση με προορισμό το 203.0.118.18 μέσω της θύρας 22, ενώ για κάθε άλλη περίπτωση δεν γίνεται η μετάφραση.

6.13) PC2: **ssh lab@203.0.118.18**

Συνδέεται στο SRV1. Η διαδρομή των πακέτων IP είναι: PC2 → FW1 → R1 → FW1 →(NAT) SRV1

SRV1 → FW1 → R1 → FW1 →(NAT) PC2, επιβεβαιώνεται από το tcpdump -vvv -e στα PC2, R1, SRV1.

6.14) Firewall > NAT > Outbound > Καταργούμε το outbound NAT για το PC1 (192.168.1.2/32).

R1: ping 192.168.1.2

Όχι, γράφει No route to host και από καταγραφή βλέπουμε πως λαμβάνει απάντηση από το 192.0.2.2 και από το 203.0.118.15 πως είναι unreachable. Δεν υπάρχει πλέον default NAT στο FW1 και δεν ορίζεται στο outbound η NAT αντιστοίχιση. Άρα και ο R1 απαντάει στο PC1 στην 192.168.1.2 επειδή δεν υπάρχει διαδρομή για αυτή.

6.15) Firewall > NAT > Outbound > Καταργούμε το advanced outbound NAT → Save.

Το ping τώρα πετυχαίνει αφού κάνει mapping αυτόματα για την IP του PC1.

6.16) R1: **ssh lab@203.0.118.18**

Ναι μπορούμε και συνδεόμαστε στο SRV1.

PC1: ssh lab@203.0.118.18

Όχι δεν μπορούμε

PC2: ssh lab@203.0.118.18

Όχι δεν μπορούμε

6.17) SRV1: **tcpdump -vvv -en -i em0**

R1: tcpdump -vvv -en -i em0

PC2: ssh lab@203.0.118.0

Στο R1 ψάχνει το 203.0.118.18 και στο SRV1 ψάχνει το 172.22.1.2. Η διεύθυνση του μεταφράζεται σε αυτή του FW1, δηλαδή source ip 192.168.1.3 → 192.0.2.1 και προωθεί πακέτο στην default, δηλαδή τον R1. Στη συνέχεια, ο R1 έχει route ήδη μέσω του FW1 και στέλνει και ενεργοποιείται ο inbound rule και το στέλνει στο SRV1. Το SRV1 την αποδέχεται προς το 192.0.2.1 και μετά το FW1 του στέλνει μήνυμα reset.

6.18) Από τον 5.35 αποτρέπουμε τη κίνηση από το DMZ στο LAN1 και όταν ο SRV1 αποδέχεται τη σύνδεση, στέλνει μήνυμα στο 192.0.2.1 το οποίο δεν βρίσκεται στο LAN1. Από τον 5.41 απαγορεύεται η κίνηση των πακέτων από 192.168.1.3 προς 172.22.1.2. Συνεπώς προκύπτει πως δεν επιτρέπεται η παροχή NAT για μετάφραση στα πακέτα που χρησιμοποιούν την IP 192.0.2.1 και στέλνονται από εσωτερικό NAT. Συνεπώς αρνείται να γίνει η μετάφραση από εσωτερικές DMZ επειδή έχει ως προορισμό την 192.0.2.1.

Άσκηση 7: IPSec site-to-site VPN

7.1) FW2: Virtual box → FW1 → Host-only (adapter 3) → cable connected → όχι connected πια → save.

7.2) FW2: Interfaces > MNG > IP Configuration/IP address = 192.168.56.3/24 → Save

7.3) Virtual box → FW1 → Host-only (adapter 3) → cable connected → connected πια → save.

7.4) Ναι μπορούμε

7.5) FW2: System> General setup > Hostname: **fw2** → **save** (στο τέλος της σελίδας)

7.6) FW2: Interfaces > WAN > IP address 192.0.2.5/30, Gateway: 192.0.2.6 → (tick) Block private networks → save

7.7) FW2: Interfaces > LAN > 192.168.2.1/24 → save

7.8) FW2: 5 → reboot

7.9) FW2: Firewall > Rules > LAN > “Add new rule” → Action: pass, Interface: LAN, Protocol: any, Source: any, Source port range: any, Destination IP: any, Destination port range: any. No fragments, not Log → **Save** → **Apply changes**

7.10) FW2: Firewall > Rules > WAN > “Add new rule” → Action: pass, Interface: WAN, Protocol: ICMP, Source: any, Source port range: any, Destination IP: WAN address, Destination port range: any. No fragments, not Log → **Save** → **Apply changes**

7.11) PC2: **ifconfig em0 192.168.2.2/24** → **route add default 192.168.2.1**

7.12) PC1: **ping 192.0.2.5**

Ναι μπορούμε

7.13) PC2: **ping 192.0.2.1**

Ναι μπορούμε

7.14) PC1: ping 192.168.2.2

Όχι δεν μπορούμε, επειδή μεταφράζονται στη δημόσια WAN των Firewalls, στέλνουν μετά στον R1, αλλά επειδή δεν υπάρχουν στον R1 οι διαδρομές για τα LAN1/2 δεν προωθεί τα ICMP Echo Request.

7.15) FW1: VPN > IPsec: Enable IPsec → Save

“add new tunnel” →

Local Subnet: LAN subnet

Remote Subnet: 192.168.2.0/24

Remote gateway: 192.0.2.5

Pre-shared key: AthanasiosPyliotis

→ **Save** → **Apply changes**

7.16) FW1 → Firewall → Rules → IPsec VPN

Βλέπω κανόνα που επιτρέπει όλη τη κίνηση, όλα τα πρωτόκολλα ανεξαρτήτως διεύθυνσης και θύρας. Γράφει επίσης default IPsec VPN.

7.17) FW1 → Diagnostics → IPsec → Security Association Database (SAD)

Όχι δεν έχουν ορισθεί σχέσεις μεταξύ των υποδικτύων.

7.18) FW1 → Diagnostics → IPsec → Security Policy Database (SPD)

Ναι έχουν ορισθεί 2 σχέσεις, 2 πολιτικές προώθησης για την αμφίδρομη επικοινωνία.

7.19) FW2: VPN > IPsec: Enable IPsec → **Save**

“add new tunnel” →

Local Subnet: LAN subnet

Remote Subnet: 192.168.1.0/24

Remote gateway: 192.0.2.1

Pre-shared key: AthanasiosPyliotis

➔ **Save → Apply changes**

7.20) FW2 → Diagnostics → IPsec → Security Association Database (SAD)

Όχι δεν έχουν ορισθεί σχέσεις μεταξύ των υποδικτύων.

7.21) FW2 → Diagnostics → IPsec → Security Policy Database (SPD)

Ναι έχουν ορισθεί 2 σχέσεις, 2 πολιτικές προώθησης για την αμφίδρομη επικοινωνία.

7.22) PC1: **ping 192.168.2.2**

Ναι μπορούμε

7.23) PC2: **ping 192.168.1.2**

Ναι μπορούμε

7.24) FW1 → Diagnostics → IPsec → SAD

Ναι, έχουν προστεθεί εγγραφές:

192.0.2.1	192.0.2.5	ESP	0f5d7046	3des-cbc	hmac-sha1
192.0.2.5	192.0.2.1	ESP	08007497	3des-cbc	hmac-sha1

7.25) FW2 → Diagnostics → IPsec → SAD

Ναι, έχουν προστεθεί εγγραφές:

192.0.2.5	192.0.2.1	ESP	08007497	3des-cbc	hmac-sha1
192.0.2.1	192.0.2.5	ESP	0f5d7046	3des-cbc	hmac-sha1

7.26) R1: **tcpdump -vvv -en -XX -i em0**

7.27) Όχι δεν παρατηρούμε πακέτα ICMP, μόνο ESP.

7.28) Εμφανίζονται πακέτα ESP με πηγή 192.0.2.1 και προορισμό 192.0.2.5 στο PC1 → PC2 και ανάποδα στο PC2 → PC1.

7.29) Όχι δεν υπάρχει πουθενά κάποια πληροφορία για τις IP διευθύνσεις των PC1,PC2.

7.30) PC2: **ssh lab@203.0.118.18**

Ναι μπορούμε να συνδεθούμε και φαίνεται να λειτουργεί κανονικά. Ξεκινάμε από το PC2 που στέλνει by default στο FW2 στο οποίο υπάρχουν κανόνες που επιτρέπουν τη κίνηση, το οποίο στη συνέχεια προωθεί στο R1 by default ξανά και μετά στο R1 υπάρχει η διαδρομή για το 203.0.118.18. Υπάρχει inbound rule στο FW1 μεταφράζεται και στέλνεται στο SRV1. Εκείνο αποδέχεται τη σύνδεση και στη συνέχεια στέλνει πακέτα προς το 192.168.2.2 (στη προηγούμενη ήταν το 192.0.2.1).

7.31) Αυτό που βλέπουμε είναι TCP πακέτα μεταξύ των 192.0.2.5 και 203.0.118.18 με τις θύρες 20484 και 22 αντίστοιχα.

7.32) Τα πακέτα είναι κρυπτογραφημένα από το ssh πρωτόκολλο και όχι το IPSec.