

**Εργαστηριο δικτυων υπολογιστων**

**Εργαστηριακη ασκηση 10: Τειχη προστασιας (Firewalls) και NAT**



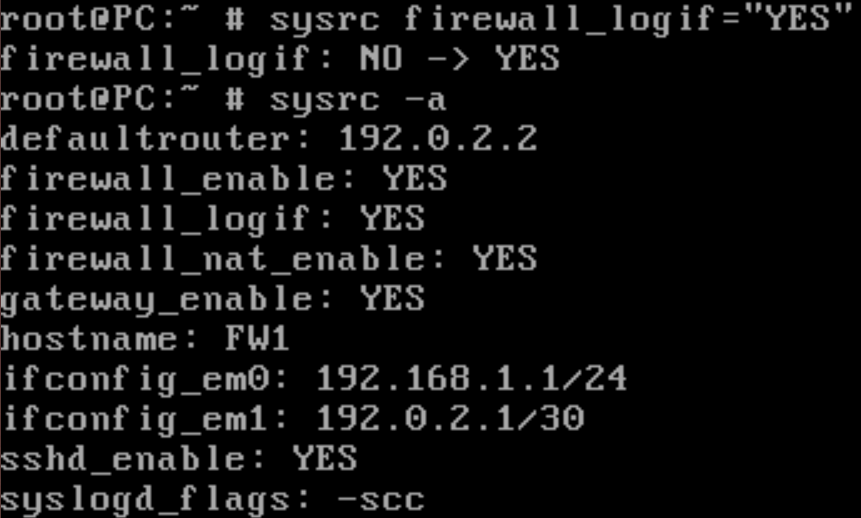


24 Μαιου, 2023

ΘΟΔΩΡΗΣ ΑΡΑΠΗΣ – EL18028

|  |  |
| --- | --- |
| **Ονοματεπώνυμο:** Θοδωρής Αράπης | **Ομάδα:** 3 |
| **Όνομα PC/ΛΣ:** DESKTOP-JGHL94V/ WINDOWS 10 | **Ημερομηνία:** 24/5/2023 |

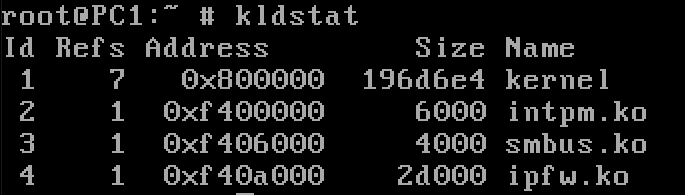
**Προετοιμασία στο σπίτι**

Παραμετροποιούμε κατάλληλα:

**Άσκηση 1: Ένα απλό τείχος προστασίας**

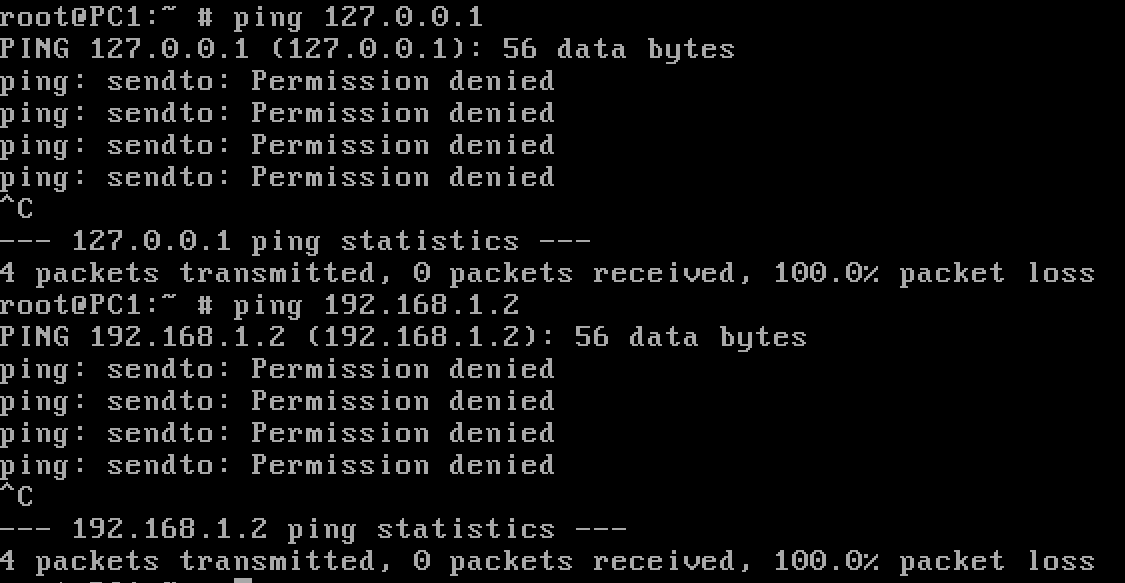
***1.1***

Εκτελούμε στο PC1 “**kldload ipfw**”.

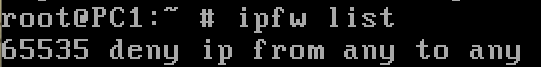
***1.2***

***ή***

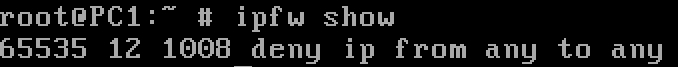
***1.3***

Όχι δε μπορούμε.

Βλέπουμε ως μήνυμα λάθους “Permission denied”.

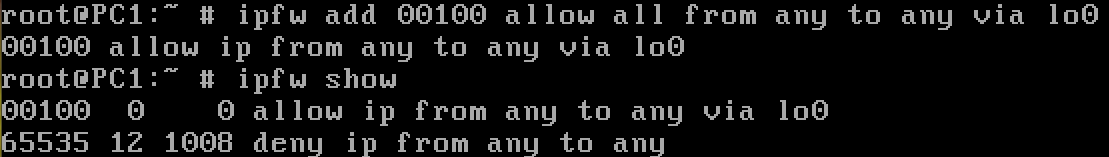
***1.4***

***1.5***

Ο παραπάνω κανόνας είναι ο προκαθορισμένος, ο οποίος απορρίπτει σιωπηλά όλα τα πακέτα. Επιπλέον, με “**ipfw show**” βλέπουμε και τις τιμές των μετρητών.

***1.6***

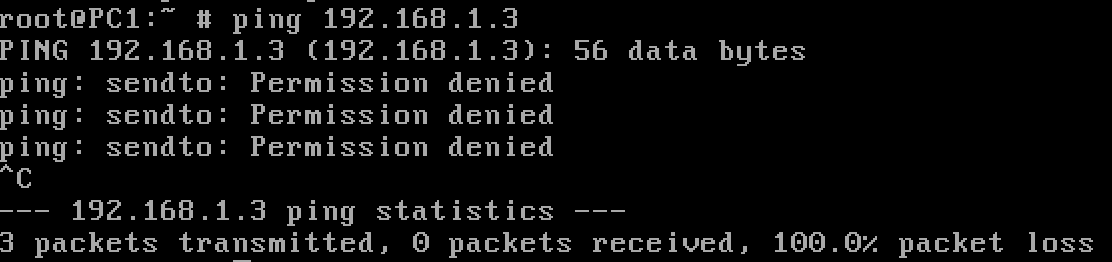
Με “**ipfw zero**”.

***1.7***

***1.8***

Ναι.

***1.9***

Όχι, παίρνουμε το ίδιο μήνυμα λάθους με πριν.

***1.10***

***1.11***

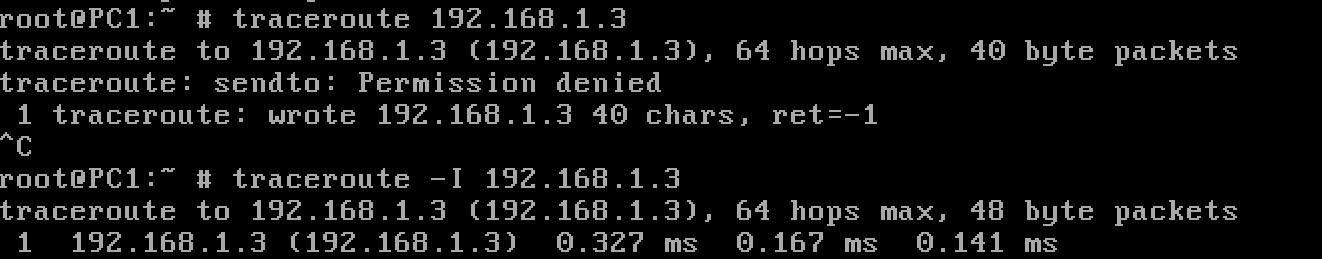
00200, 100 δηλαδή παραπάνω από το προηγούμενο, αφού δε το ορίσαμε ρητά α/α.

***1.12***

Πετυχαίνουν αμφότερα.

***1.13***

Δε μπορούμε καθώς το traceroute by default χρησιμοποιεί UDP Datagrams, τα οποία και δεν επιτρέπονται να περάσουν από το firewall μας. Αν ωστόσο εκτελέσουμε “**traceroute -I 192.168.1.3**”, ώστε να στείλουμε ICMP Echo αντ’ αυτών, τότε πετυχαίνει.



***1.14***

Εκτελούμε “**ipfw add allow udp from me to any 33434-33534**”.

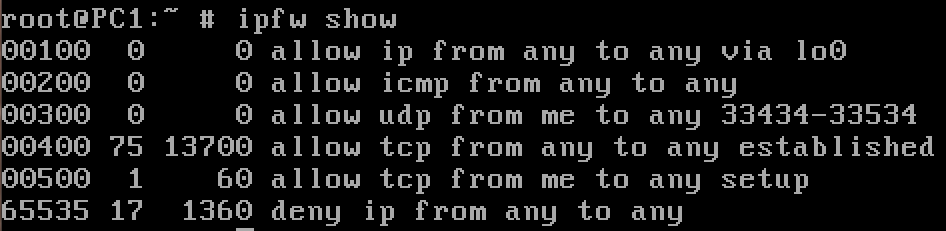
***1.15***

***1.16***

Εκτελούμε “**ipfw add allow tcp from any to any established**” και “**ipfw add allow tcp from me to any setup**”.

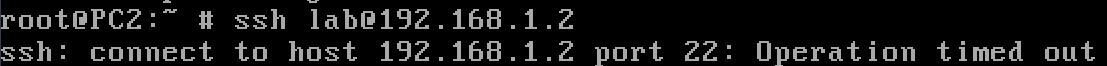
***1.17***

Εκτελούμε “**ipfw zero**” → “**ssh lab@192.168.1.3**” → “**ls**” → “**exit**”.

***1.18***

Η πρώτη στήλη μετά τον αριθμό του κανόνα (και εξαιρουμένου του τελευταίου κανόνα, του οποίου οι μετρητές δε μηδενίζονται) δείχνει πόσες φορές εφαρμόστηκε ο κάθε κανόνας στην παραπάνω διαδικασία. Άρα εφαρμόστηκε μία φορά ο κανόνας 00500 (στην τριμερή χειραψία) και 75 φορές ο κανόνας 00400 (κατά τη μεταφορά δεδομένων στη σύνδεση ssh).

***1.19***

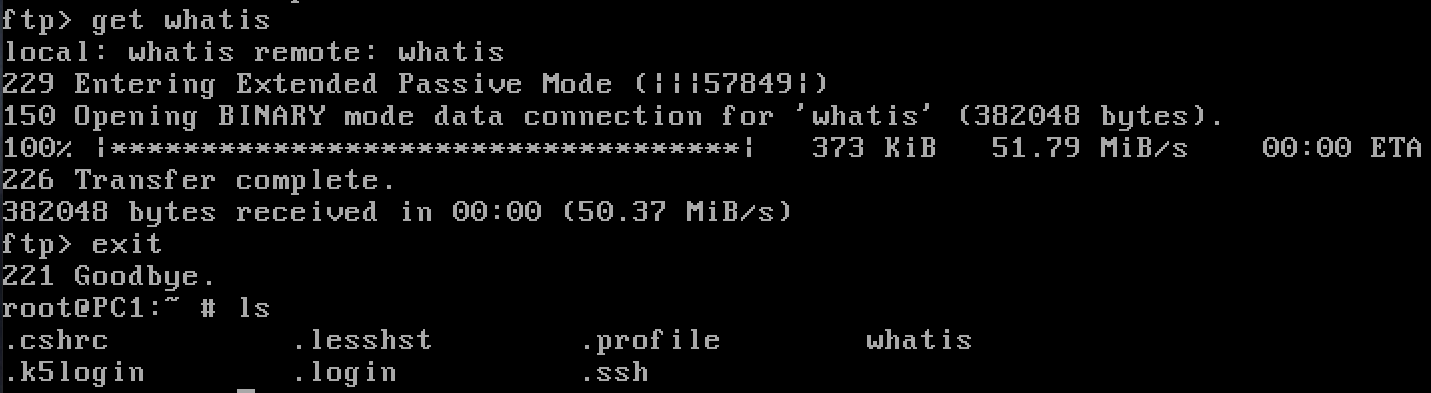
Δε μπορούμε, καθώς έχουμε επιτρέψει μόνο απερχόμενες tcp συνδέσεις από τον PC1. (00500).

***1.20***

Εκτελούμε “**service ftpd onestart**”.

***1.21***

Εκτελούμε στον PC1 “**ftp lab@192.168.1.3**”, εισάγουμε κωδικό “**ntua**”, όντας στο FTP prompt εκτελούμε “**cd /usr/bin**” → “**get whatis**”. Βλέπουμε πως το αρχείο κατέβηκε κανονικά:

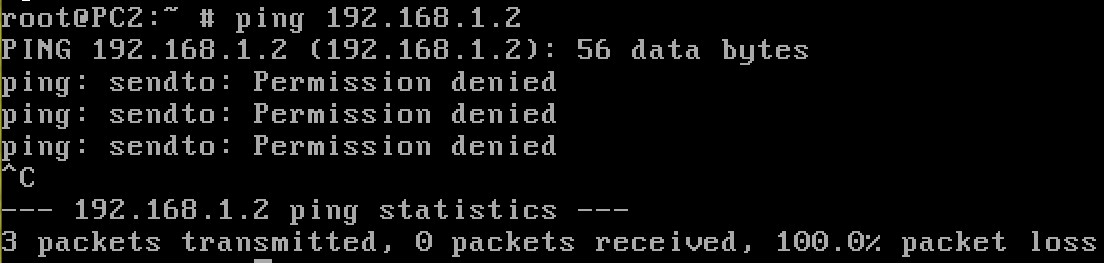


**Άσκηση 2: Ένα πιο σύνθετο τείχος προστασίας**

***2.1***

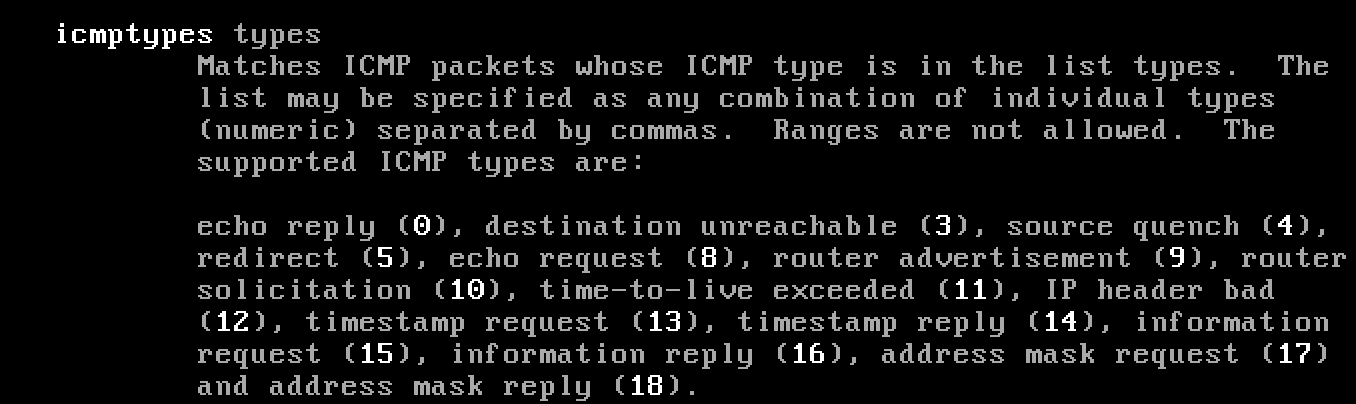
Στο PC2 “**kldload ipfw**”.

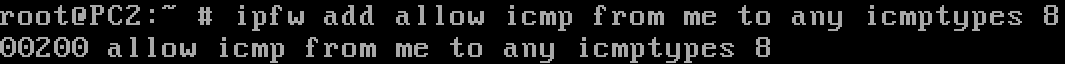
***2.2***

Όχι. (Permission denied)

***2.3***

***2.4***

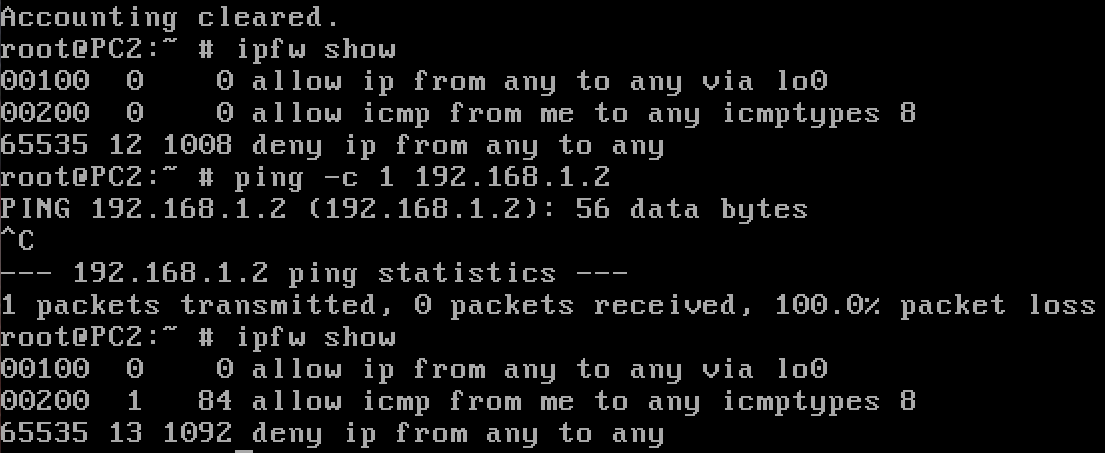
Από το man page του ipfw βρίσκουμε:

Οπότε τρέχουμε την ακόλουθη εντολή:

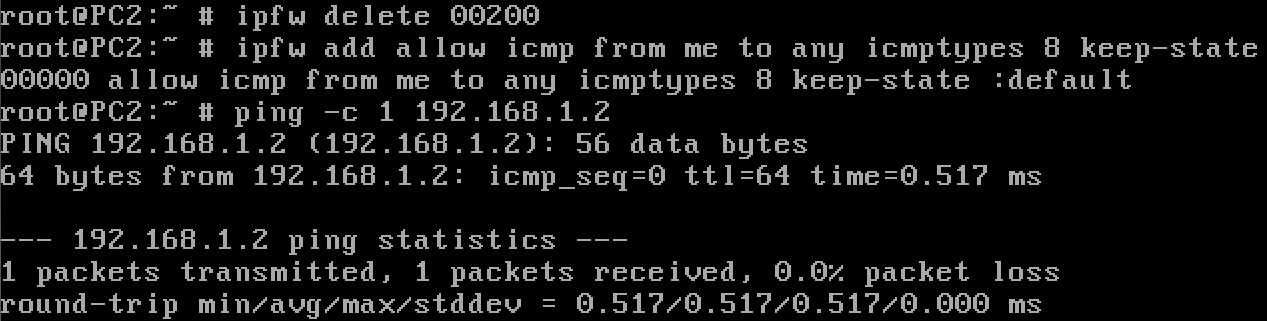
***2.5***

Όχι, αλλά δε λαμβάνουμε Permission Denied αυτή τη φορά.

***2.6***

Για να παρατηρήσουμε το φαινόμενο, αρχικά καθαρίζουμε τους μετρητές (“**ipfw zero**”), στη συνέχεια στέλνουμε ένα ICMP Echo request (“**ping -c 1 192.168.1.2**”) και μετά εκτελούμε “**ipfw show**” και βλέπουμε πως ο κανόνας 00200 χρησιμοποιείται μία φορά, επομένως τα πακέτα ICMP όταν είναι εξερχόμενα περνούν το τείχος προστασίας του PC2.

***2.7***

Ναι, πλέον μπορούμε

***2.8***

Ναι, μπορούμε.

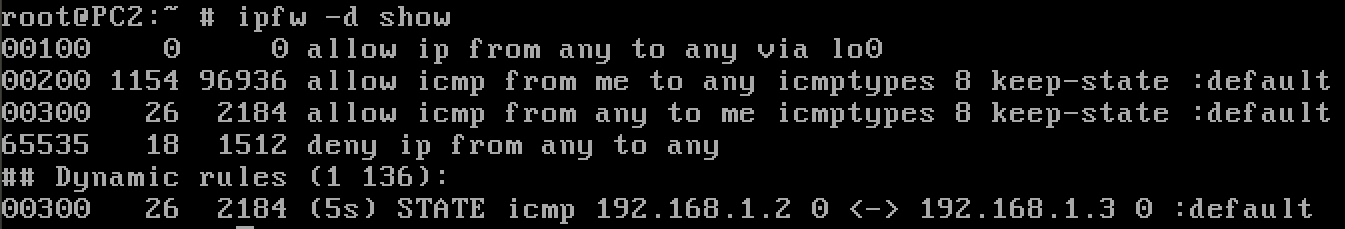
***2.9***

Όχι, πλέον δεν επιτυγχάνει. Το Ping πέτυχε προηγουμένως, καθώς η επιλογή keep-state που είχαμε προσθέσει έκανε τη σύνδεση PC1-PC2 stateful με αποτέλεσμα τα Ping του PC1 να περνάνε όσο ο PC2 έστελνε ping.

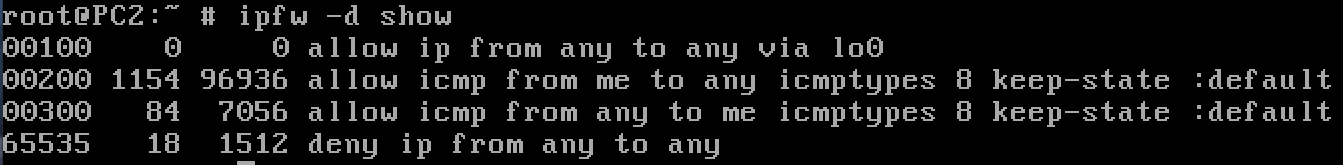
***2.10***

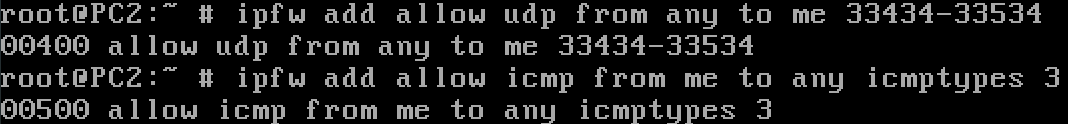
Εκτελούμε “**ipfw add icmp allow from any to me icmptypes 8 keep-state**”.

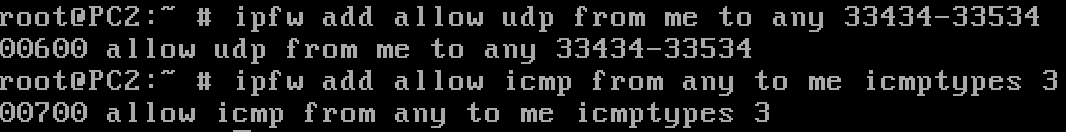
***2.11***

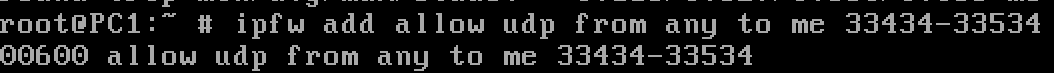
Βλέπουμε τη χρήση ενός δυναμικού κανόνα κατά την επικοινωνία.

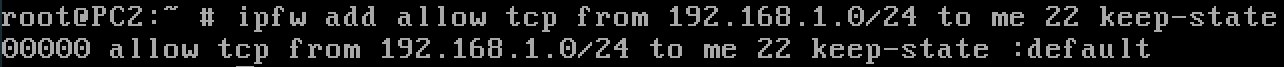
***2.12***

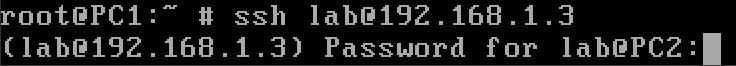
Πλέον βλέπουμε μόνο τους στατικούς κανόνες:

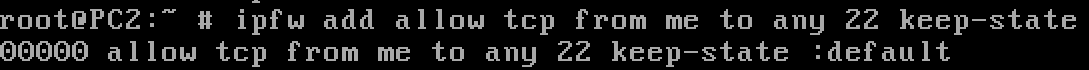
***2.13***

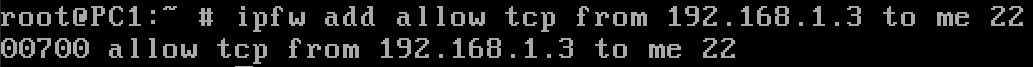
***2.14***

***2.15***

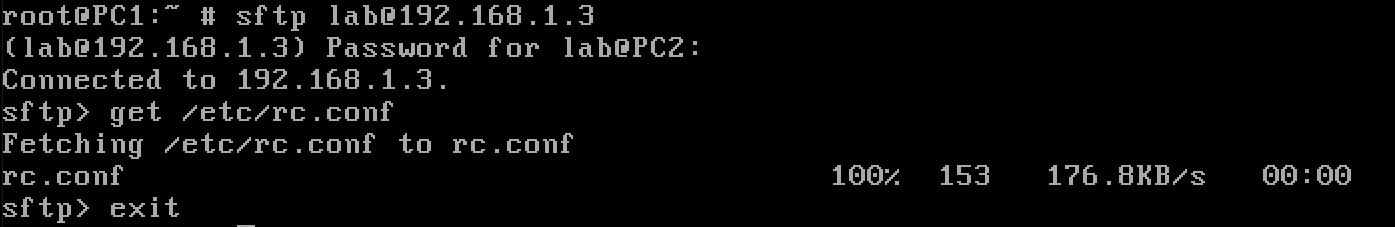
***2.16***

***2.17***

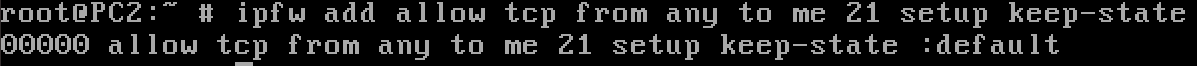
***2.18***

***2.19***

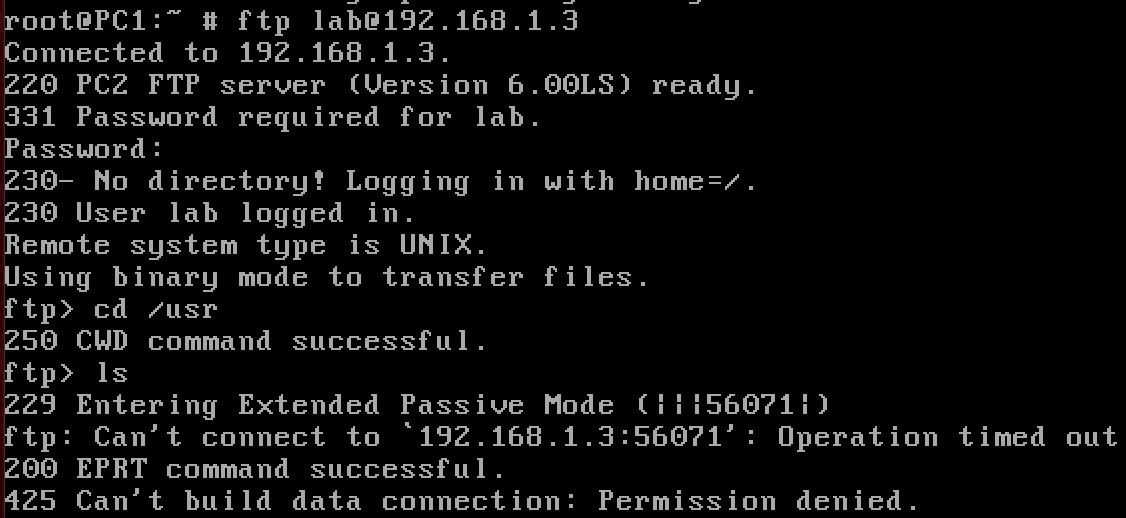
***2.20***

Ναι, αφού το sftp τρέχει πάνω από ssh session.

***2.21***

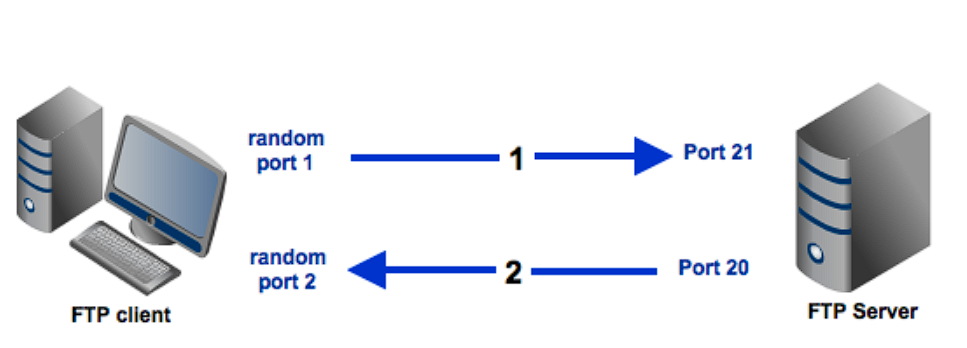
Δε μπορούμε, οπότε εισάγουμε τον παρακάτω κανόνα:

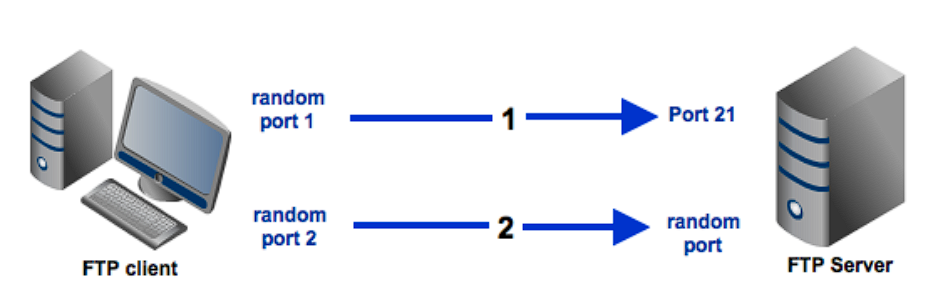
***2.22***

Έχουμε ενεργοποιήσει μόνο την θύρα 21, η οποία αφορά συνδέσεις Control FTP και όχι την 20 που αφορά FTP data transfer (το οποίο συμβαίνει με την εντολή ls).

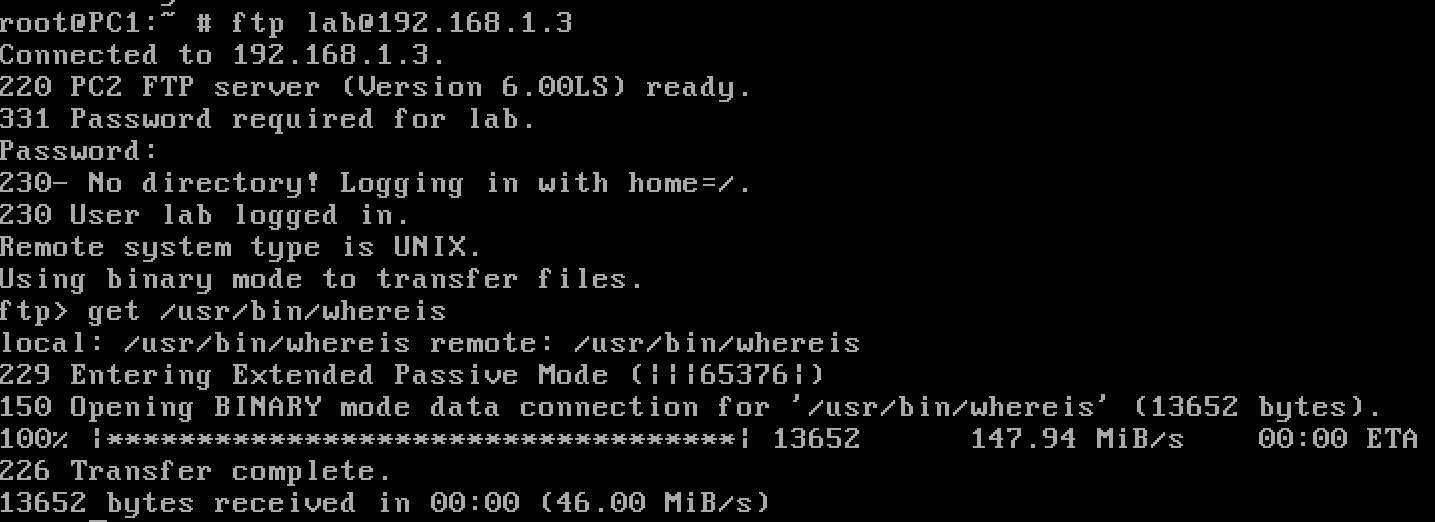
***2.23***

Τον κανόνα “**ipfw add allow tcp from any 1024-65535 to me 1024-65535 setup keep-state**”, βάσει και του παρακάτω σχήματος.

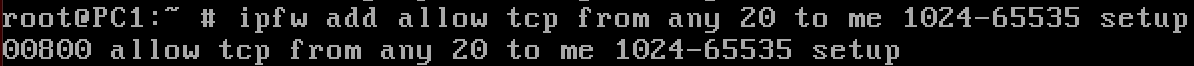
 

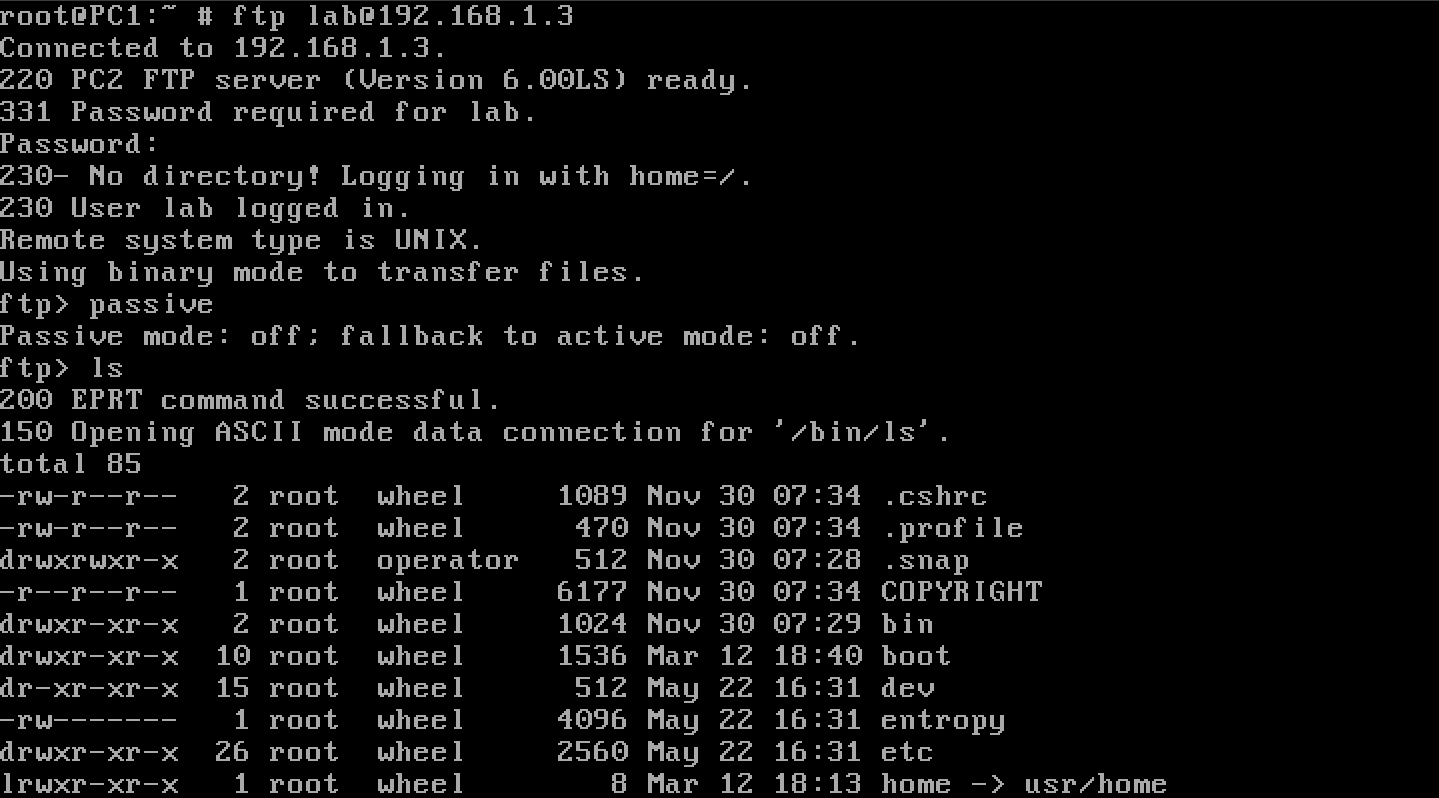
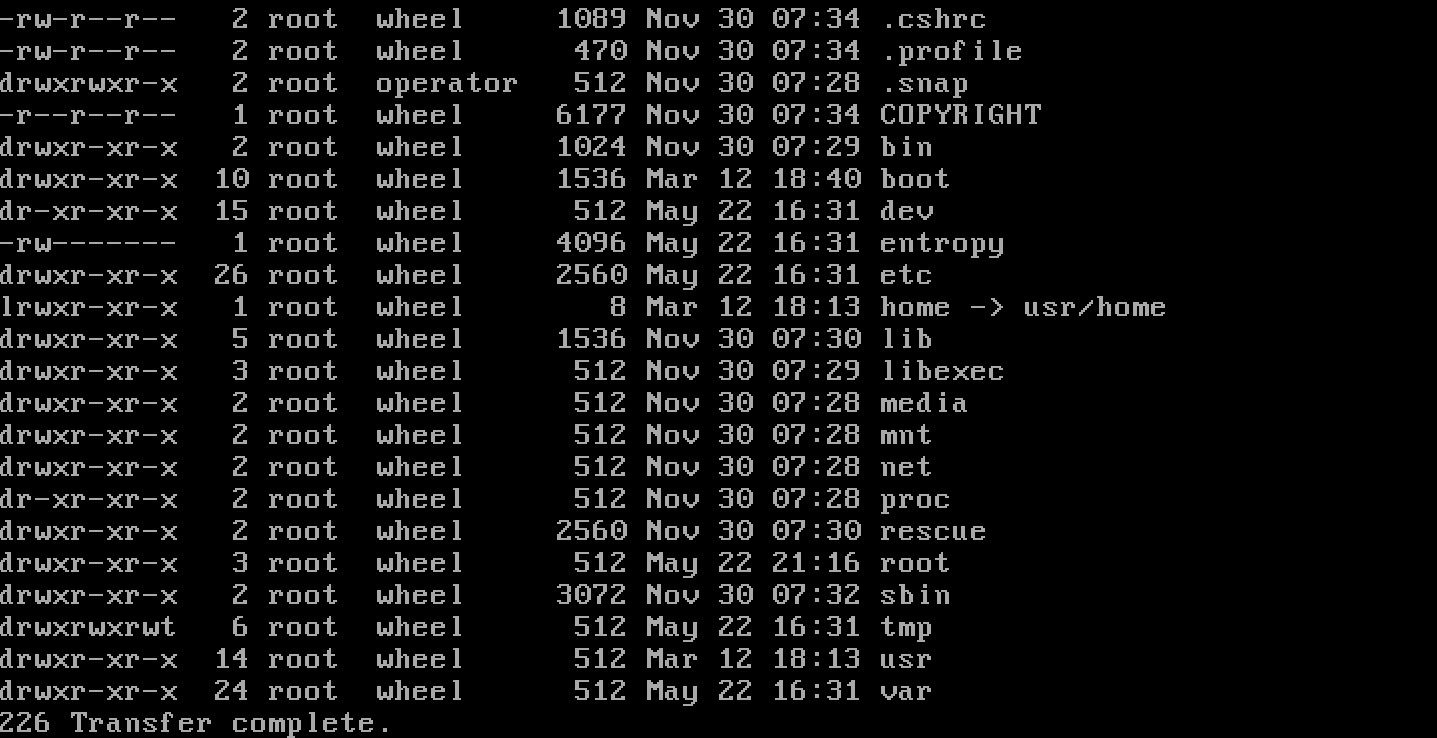
 

***2.24***

Ναι.

***2.25***

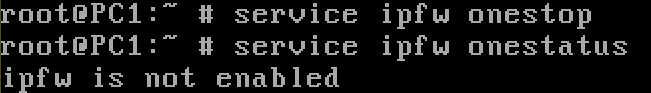
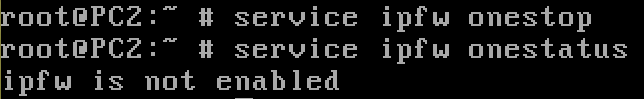
Εισάγουμε τα παρακάτω στα PC1 και PC2 αντίστοιχα και βλέπουμε πως επιτυγχάνει.



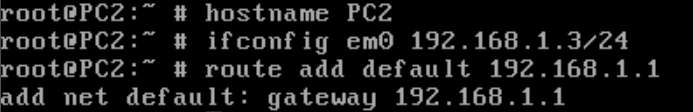
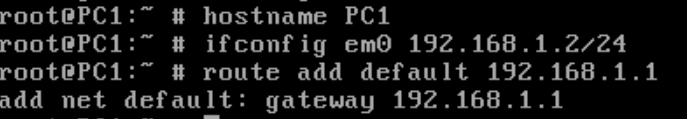
***2.26***

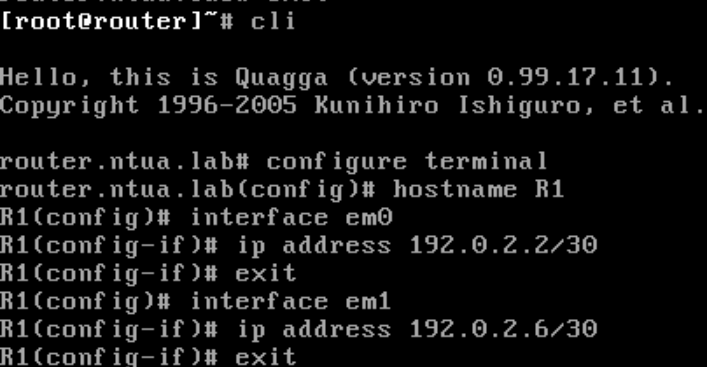
Βλέπουμε πως το ftp μπορεί να αξιοποιεί μεγάλο εύρος θυρών, με αποτέλεσμα εάν κάποιος θέλει να αφήνει ενεργή την υπηρεσία να εκτίθεται σε κίνδυνο λόγω των πολλών ανοιχτών θυρών. Για αυτό θα μπορούσαμε να αξιοποιήσουμε π.χ. δυναμικούς κανόνες, ώστε να επιτρέπεται ανταλλαγή δεδομένων μόνο αφού έχει εγκατασταθεί η σύνδεση.

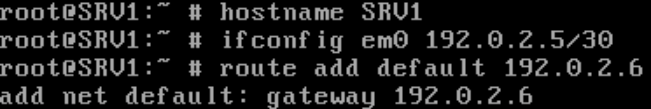
***2.27***

Εκτελούμε στα PC1, PC2 “**service ipfw onestop**”.

**Άσκηση 3: Απλό Network Address Translation**

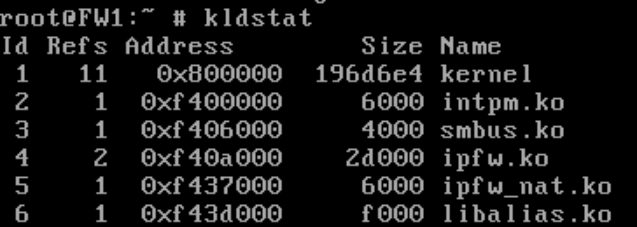
***3.1***

***3.2***

***3.3***

***3.4***

Εκτελούμε στα μηχανήματα “**service ftpd onestart**”.

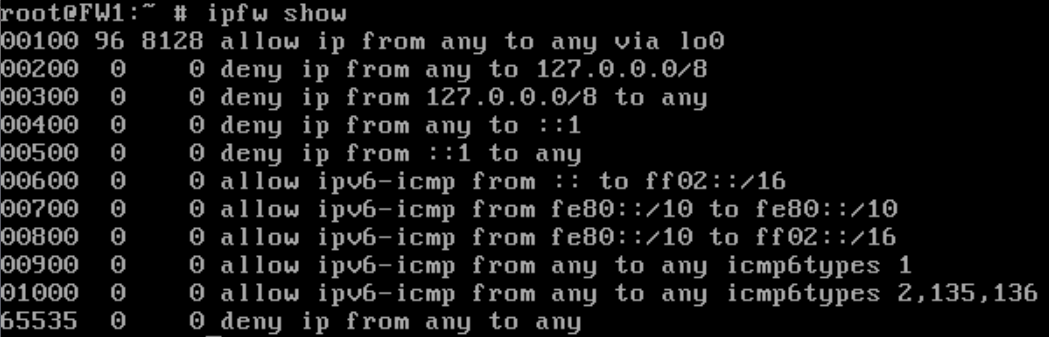
***3.5***

***3.6***

Το ipfw.

***3.7***

***3.8***

Βλέπουμε τους παρακάτω 11 κανόνες, με τον τελευταίο να αποτελεί τον default, ο οποίος απορρίπτει σιωπηλά όλα τα πακέτα.

***3.9***

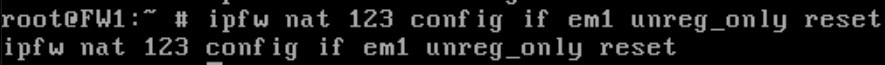
Με την εντολή “**ipfw nat show config**” και βλέπουμε πως δεν υπάρχει κανένας πίνακας.

***3.10***

Όχι, σε καμία από τις 2.

***3.11***

Όχι.

***3.12***

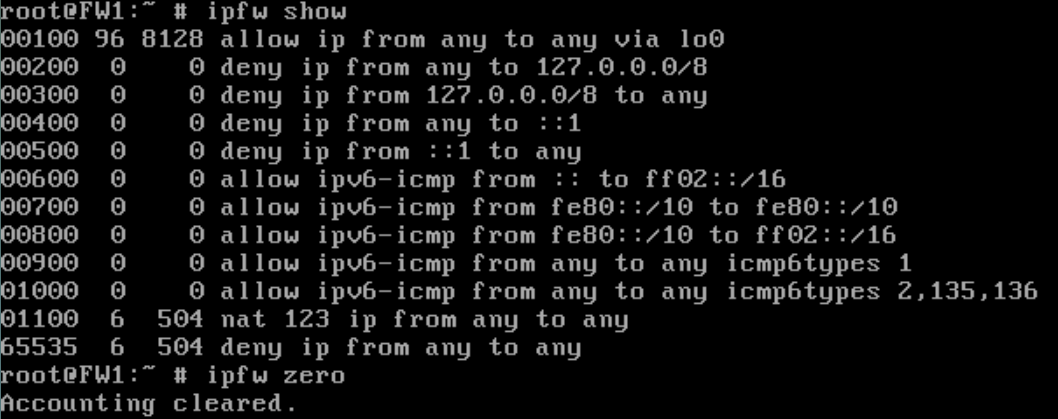
***3.13***

***3.14***

Ναι, μπορούμε και στις 2.

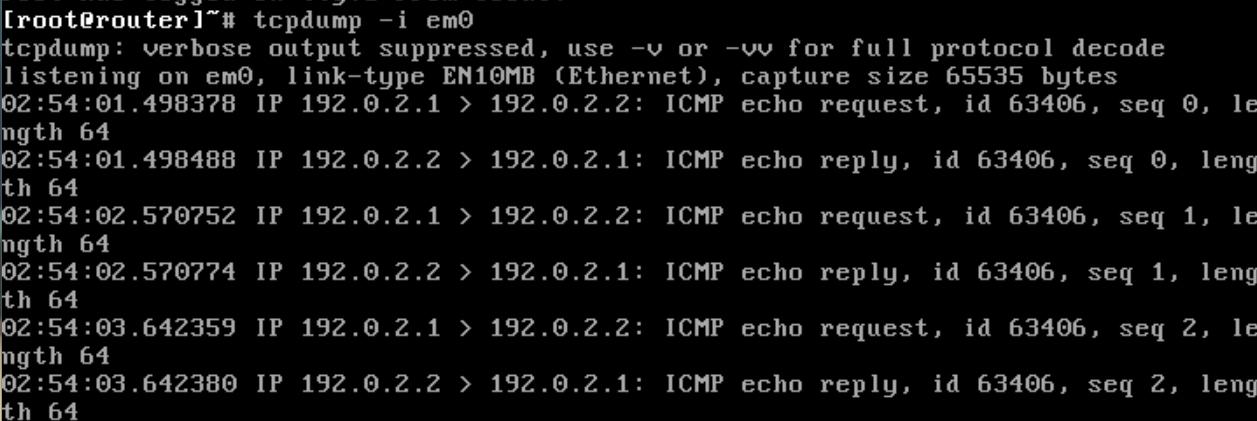
***3.15***

Εκτελούμε στο R1 “**tcpdump -i em0**”

***3.16***

***3.17***

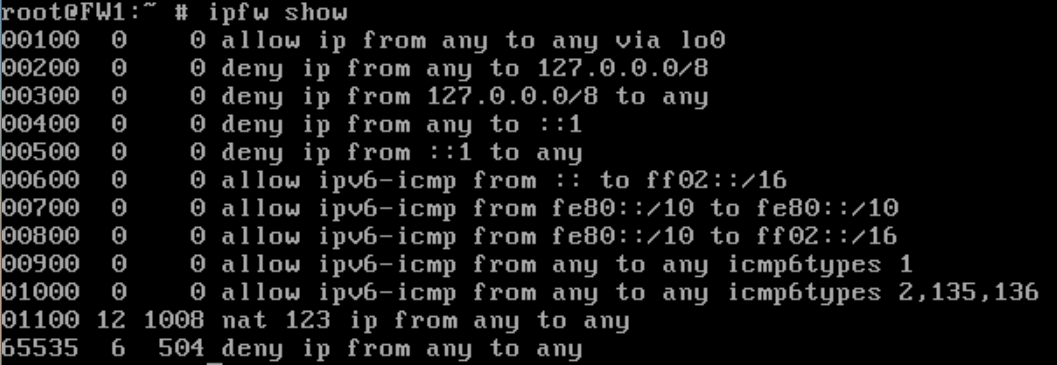
Πηγή των ICMP Echo requests εμφανίζεται να είναι η 192.0.2.1, δηλαδή η em1FW1.



***3.18***

Διεύθυνση προορισμού των ICMP Echo reply είναι η 192.0.2.1 (em1FW1).

***3.19***

Υπεύθυνος είναι ο κανόνας “**nat 123 ip from any to any**”.

***3.20***

Βλέπουμε πως εφαρμόστηκε 12 φορές. Συνολικά πέρασαν από το τείχος 6 πακέτα (3 requests και 3 reply), ωστόσο, το κάθε πακέτο μπήκε για μετάφραση κατά την είσοδο και κατά την έξοδό του από αυτό, οπότε και προκύπτει το 12.

***3.21***

Ναι μπορούμε.

***3.22***

Είναι ο ίδιος κανόνας με παραπάνω, ο οποίος χρησιμοποιήθηκε 2 φορές αυτή τη φορά, για 2 echo request.

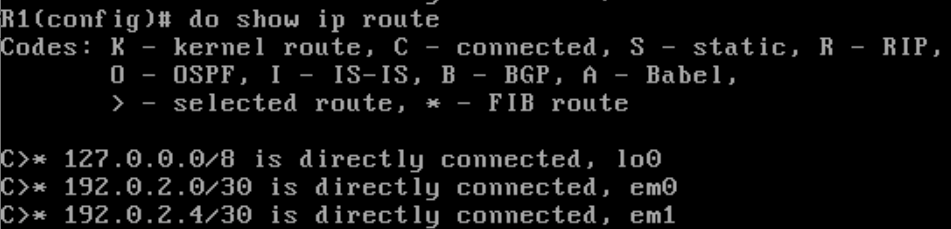
***3.23***

Ωθείται μεν για μετάφραση, αλλά δεν υπόκειται σε μετάφραση.

***3.24***

Ναι.

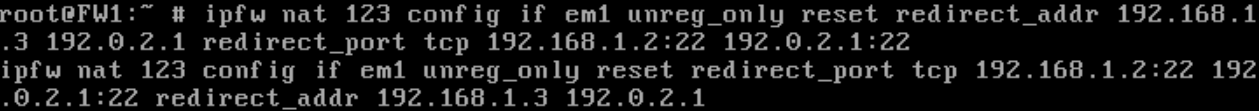
***3.25***

Κάνοντας “**tcpdump -i em1**” βλέπουμε πως ο R1 απαντάει με “host 192.168.1.3 unreachable”, ενώ δε περνάει τίποτα από τον R1 στο WAN1, επομένως είναι πρόβλημα δρομολόγησης, καθώς βλέποντας και τον πίνακα δρομολόγησης του R1 παρατηρούμε πως δεν έχει κατάλληλη εγγραφή για να απαντήσει στο PC2.

***3.26***

***3.27***

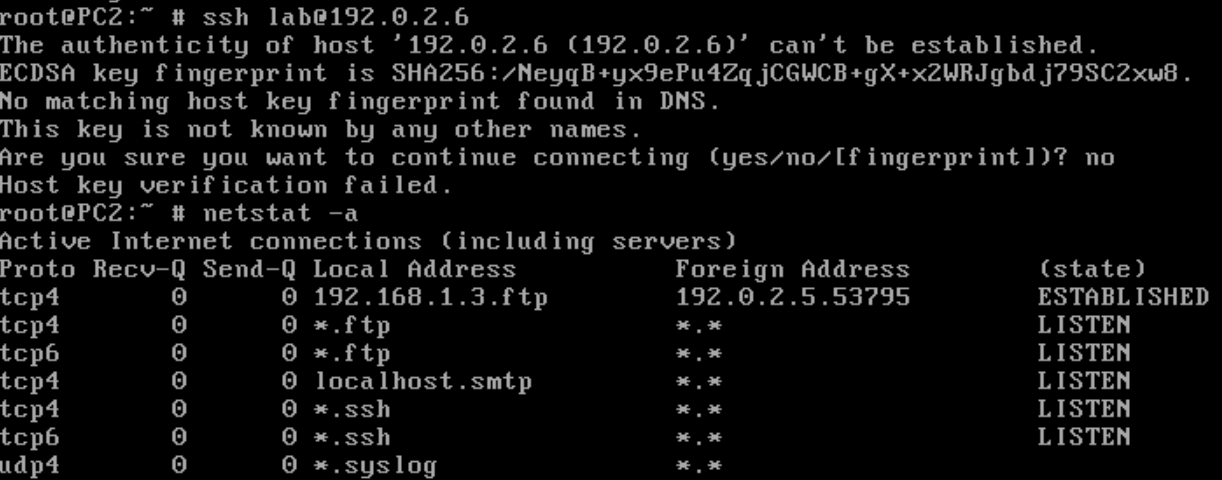
Ναι είναι επιτυχής (“**ssh lab@192.0.2.1**” από το SRV1) και βλέπουμε από το prompt για το password πως έχουμε συνδεθεί στο PC2.

***3.28***

***3.29***

Τώρα συνδεθήκαμε στο PC1 και το βλέπουμε από το prompt.

***3.30***

Εκτελούμε στα PC1 και PC2 “**netstat -a**” και βλέπουμε στο PC2 πως έχει γίνει σύνδεση ftp, επομένως εκεί συνδέθηκε ο SRV1.

***3.31***

Ναι μπορούμε.

***3.32***

Το PC2.

***3.33***

Στο PC1.

**Άσκηση 4: Τείχος προστασίας και NAT**

***4.1***

Όχι, και τα 2 ping αποτυγχάνουν.

***4.2***

Ναι και τα 2 γίνονται αποδεκτά. Αποτυγχάνουν, ωστόσο, αφού απενεργοποιήσαμε το one-pass, οπότε και ελέγχθηκε ο επόμενος κανόνας, ο οποίος εν προκειμένω ήταν ο προκαθορισμένος που απέρριψε τα πακέτα.

***4.3***

***4.4***

Ναι, σε αμφότερες τις διεπαφές.

***4.5***

Στο FW1.

***4.6***

Ο κανόνας που εισάγαμε στο 4.3.

***4.7***

***4.8***

***4.9***

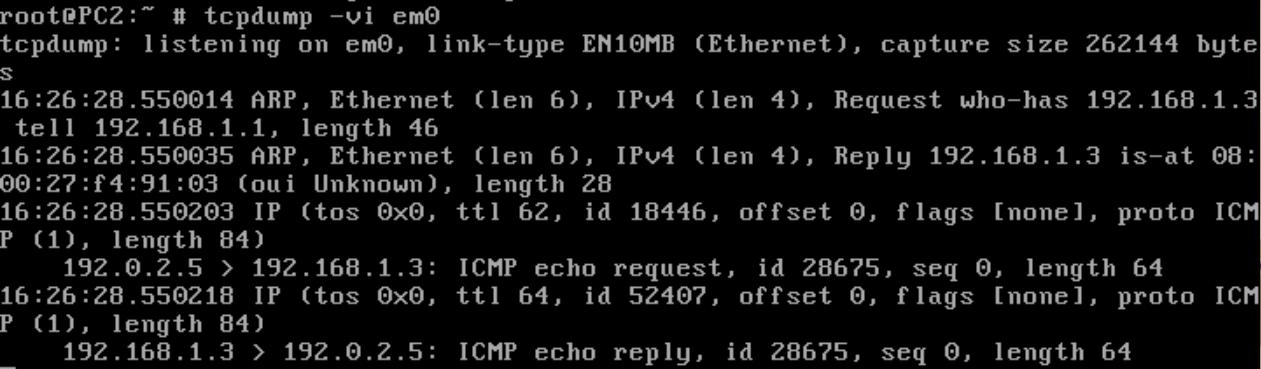


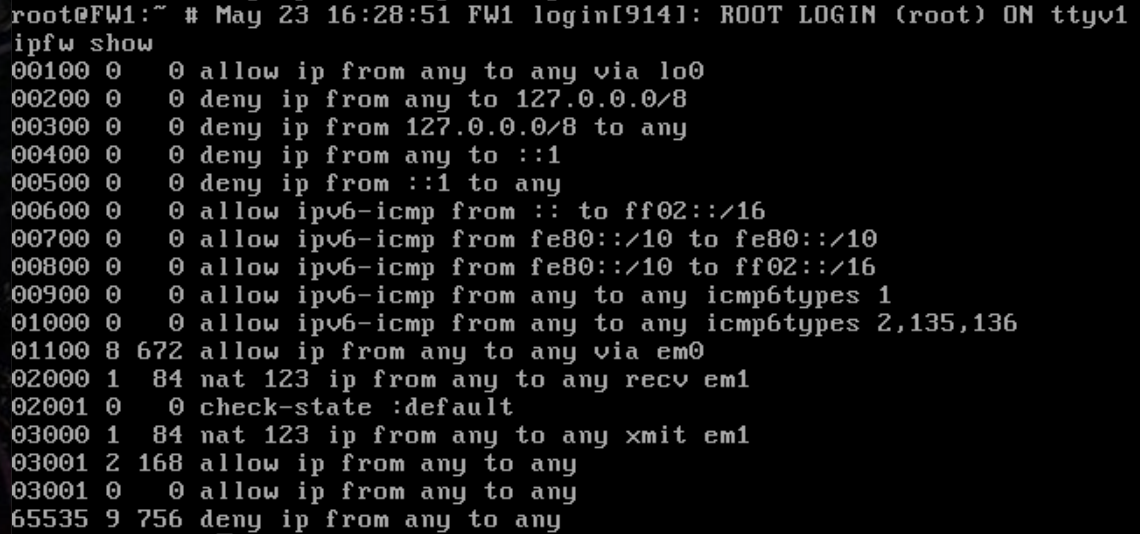
***4.10***

***4.11***

Το FW1.

***4.12***

Το PC2. Παρακάτω βλέπουμε το tcpdump στο PC2.

Όπως βλέπουμε παρακάτω, ο κανόνας 01100 εφαρμόστηκε 6 φορές για τα 3 ICMP Echo request που στείλαμε από το PC1 στο 192.0.2.1, 2 φορές εφαρμόστηκε ο ίδιος κανόνας για το ping από το SRV1 στο PC2 και από μία φορά οι κανόνες 02000 και 03000 για το ίδιο ping.

***4.13***

Στο FW1.

***4.14***

Στο PC1

***4.15***

Στο PC2.

***4.16***

Ναι.

***4.17***

Ναι.

***4.18***

Ναι.

***4.19***

***4.20***

Επιτυγχάνουν μόνο τα 4.11 και 4.13, καθώς όλα τα άλλα απαιτούν να εισέλθει κίνηση από το WAN1 μέσω του firewall, πράγμα που απαγορεύσαμε.

***4.21***

***4.22***

Ναι.

***4.23***



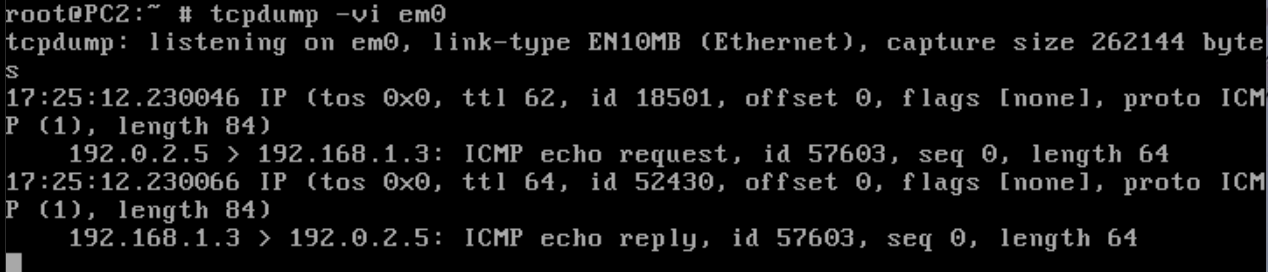
***4.24***

Ναι.

***4.25***



***4.26***

Το PC2, όπως βλέπουμε με “**tcpdump -vi em0**” στο FW1.

***4.27***

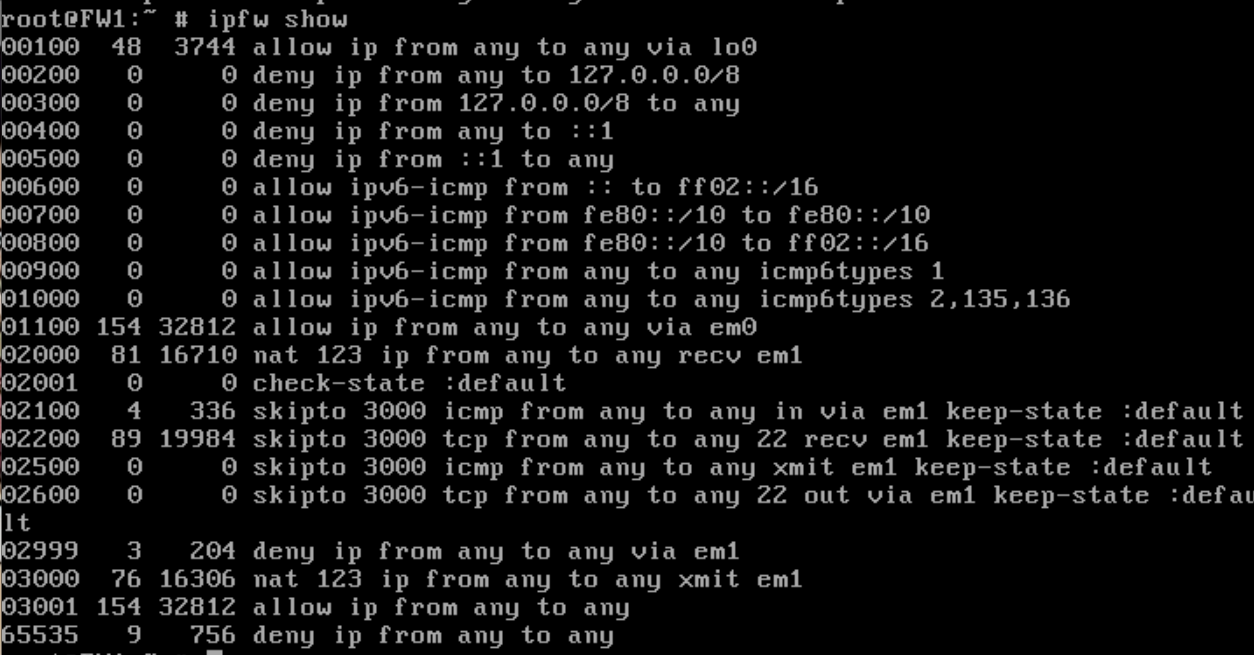
Εκτελούμε “**ipfw add 2200 skipto 3000 tcp from any to any 22 recv em1 keep-state**”.

***4.28***

Στο PC1.

***4.29***

Όχι, καθώς απορρίπτεται από τον κανόνα 2999.



***4.30***

Εισάγουμε τους κανόνες “**ipfw add 2300 skipto 3000 tcp from any to any 21 setup recv em1 keep-state**” και “**ipfw add 2700 skipto 3000 tcp from any 20 to any setup xmit em1 keep-state**”.

**Άσκηση 5: Τείχος προστασίας με γραφικό περιβάλλον διαχείρισης**

***5.1***

192.168.1.1/24

***5.2***

10.0.0.1/30.

***5.3***

87%

***5.4***

Τις αναμενόμενες 4.

***5.5***

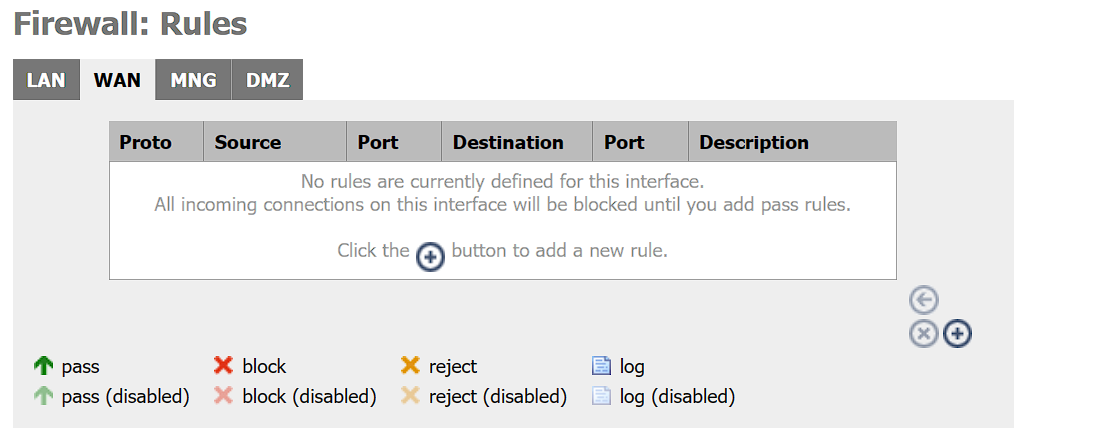
172.22.1.1/24.

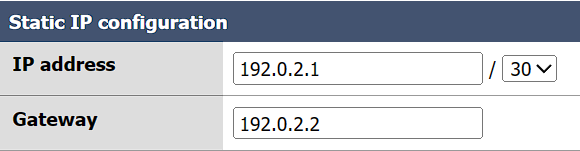
***5.6***

***5.7***

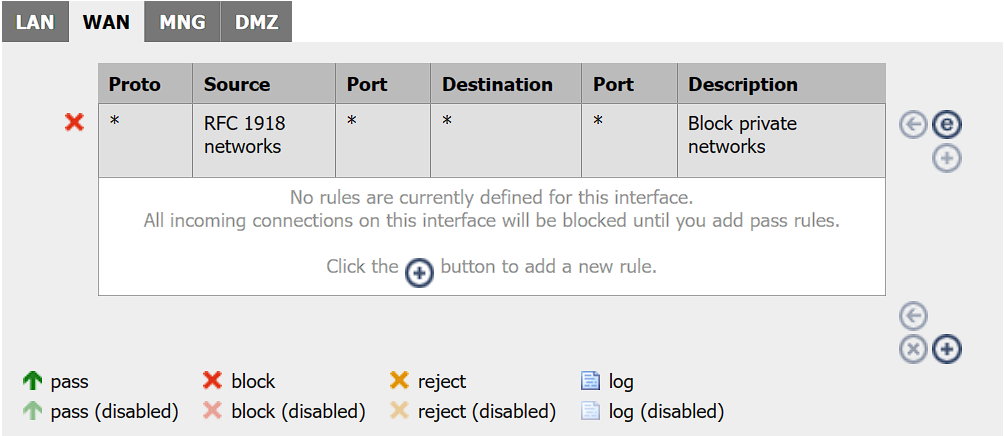
Κάνουμε την αλλαγή.

***5.8***

Δεν υπάρχουν κανόνες που να έχουμε ορίσει, ωστόσο by default όλες οι εισερχόμενες συνδέσεις σε αυτή τη διεπαφή θα μπλοκάρονται μέχρι να βάλουμε pass rules.

***5.9***

***5.10***

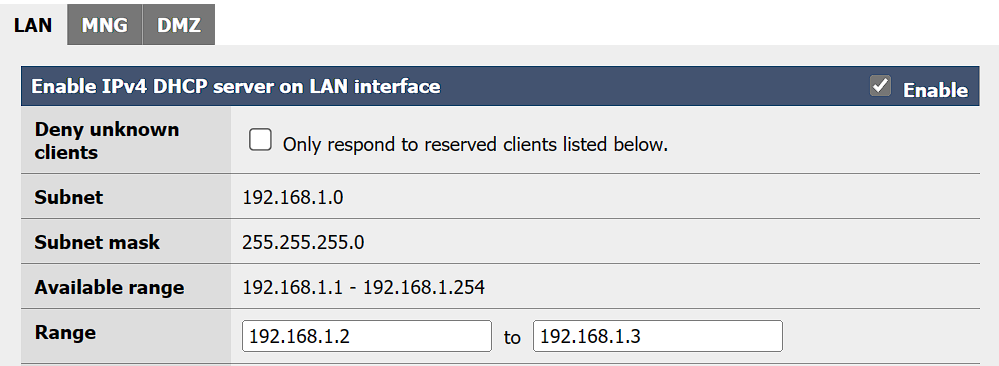
Ναι, υπάρχει ο παρακάτω κανόνας:

***5.11***

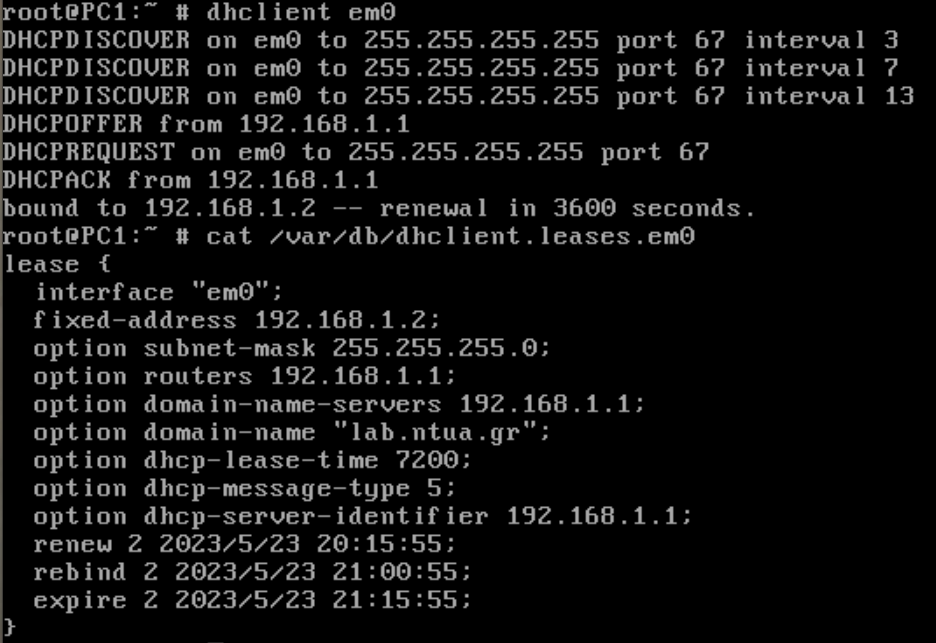
Όχι, καμία.

***5.12***

Την ενεργοποιούμε.

***5.13***

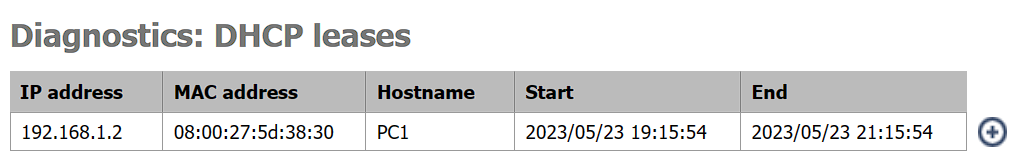
***5.14***

IP: 192.168.1.2, Default Gateway: 192.168.1.1, DNS server: 192.168.1.1.

***5.15***

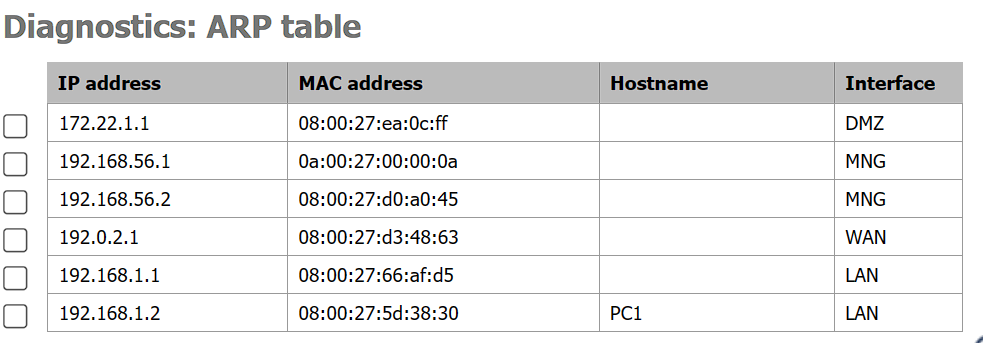
Προκειμένου να χρησιμοποιηθεί η διεπαφή του FW1 στο LAN1 ως DNS για τους πελάτες DHCP.

***5.16***

Στο “dhcp leases”.

***5.17***

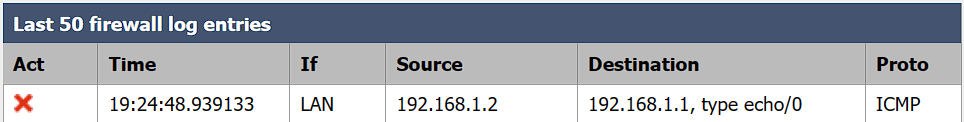
Τις παρακάτω 6:



***5.18***

Όχι.

***5.19***

Βλέπουμε το αποτυχημένο ping.

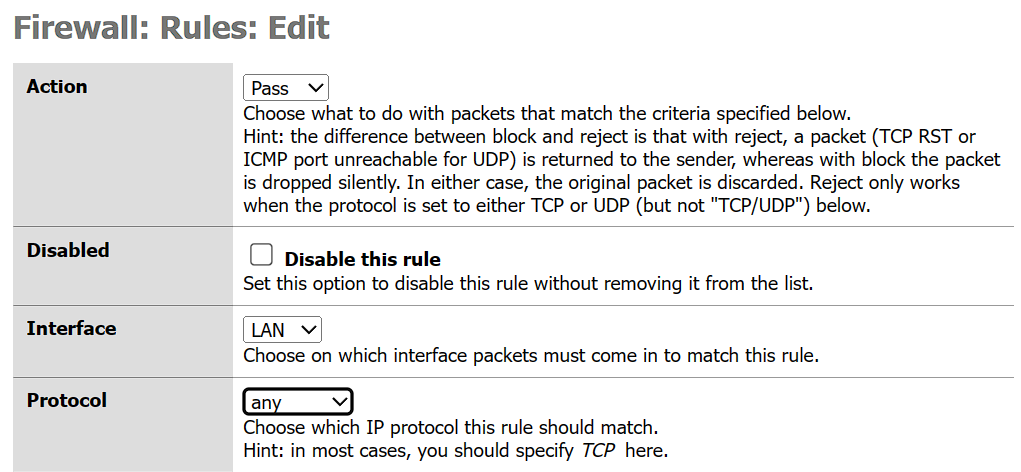
***5.20***

Τα εξής 4:

***5.21***

Κανέναν.

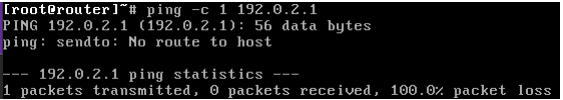
***5.22***

Ορίζουμε τις παρακάτω επιλογές

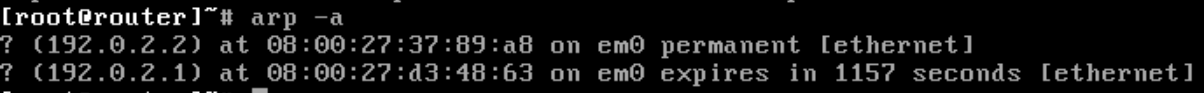
***5.23***

Ναι.

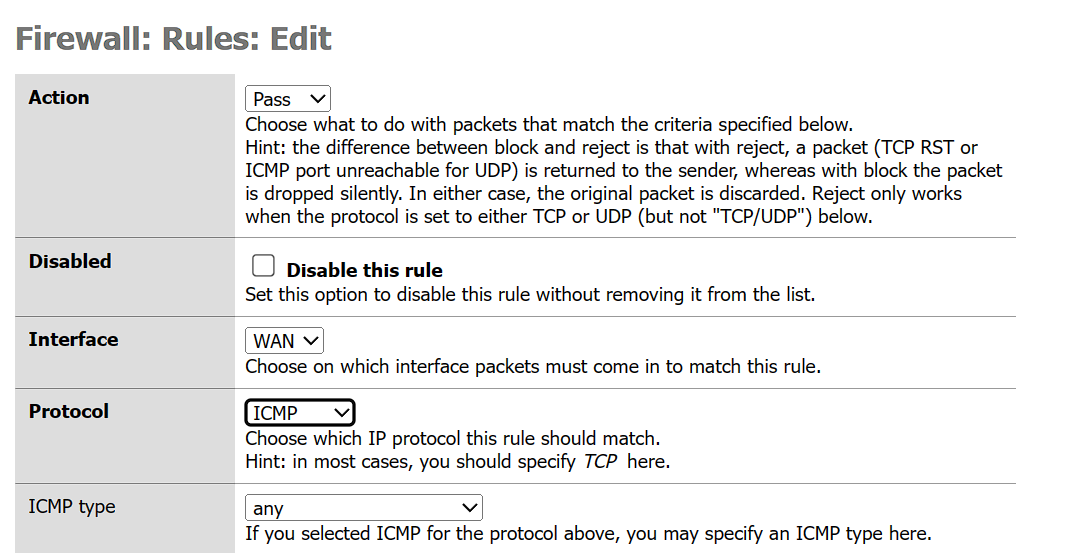
***5.24***

Όχι.

***5.25***

Ναι.

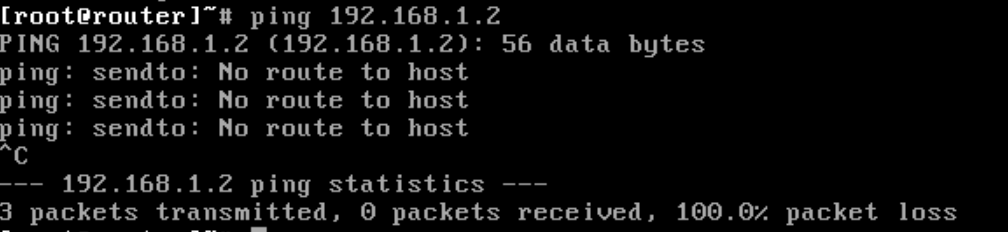
***5.26***

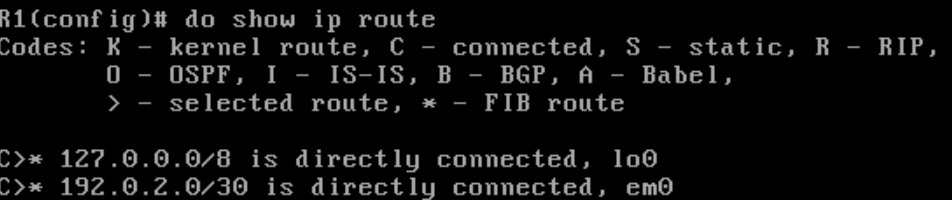


***5.27***

Ναι.

***5.28***

Όχι δε μπορούμε, καθώς ο R1 δεν έχει ούτε default gateway, ούτε κατάλληλη εγγραφή για το δίκτυο του PC1.



***5.29***

Ναι μπορούμε, αφού το PC1 έχει default gateway και επιπλέον το NAT είναι by default ενεργοποιημένο, επομένως λόγω των stateful κανόνων μπορεί το R1 να απαντήσει.

***5.30***

Όχι, καθώς ο SRV1 δε μπορεί να δρομολογήσει την απάντηση.

***5.31***

***5.32***

Ναι.

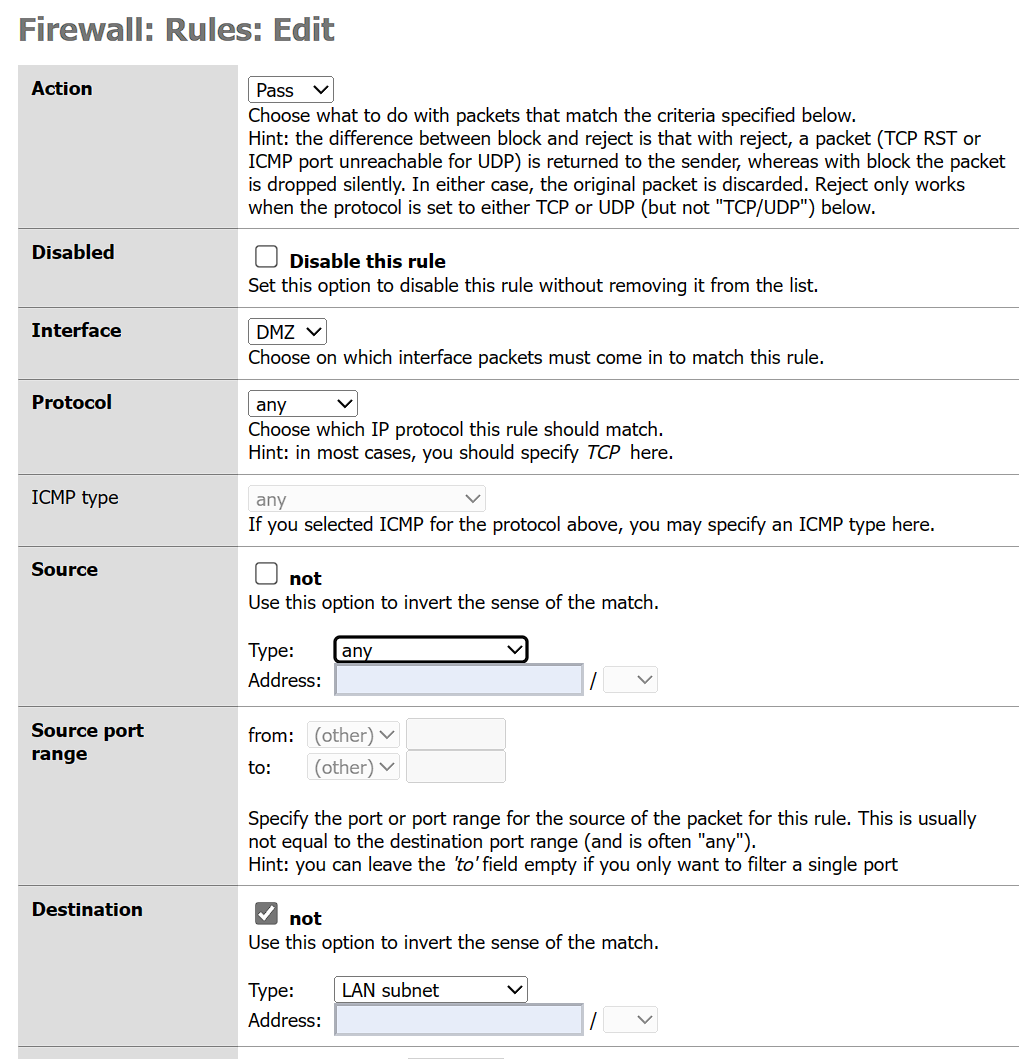
***5.33***

Όχι. Δεδομένου πως δεν έχουμε προσθέσει κανόνες στο firewall για το DMZ, όλα τα πακέτα μπλοκάρονται, ενώ προηγουμένως στο 5.32 μπορούσαμε αφού οι κανόνες είναι stateful, οπότε αφού επιτρεπόταν κίνηση από το PC1 προς τον SRV1, επιτρεπόταν και η αντίστροφη

***5.34***

Όχι, για τον ίδιο λόγο με το 5.33.

***5.35***

Κάνουμε τις αλλαγές.

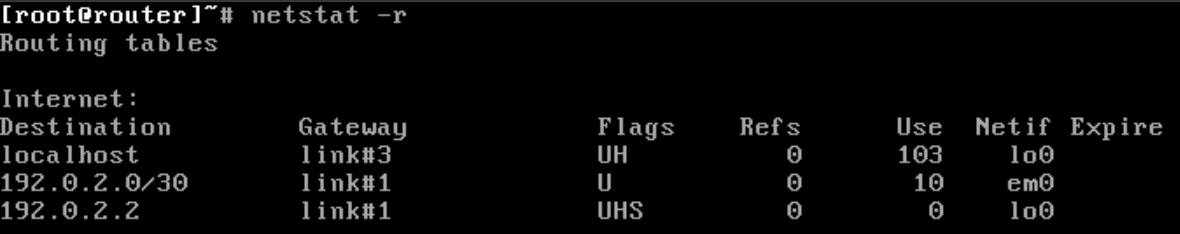
***5.36***

Ναι.

***5.37***

Ναι.

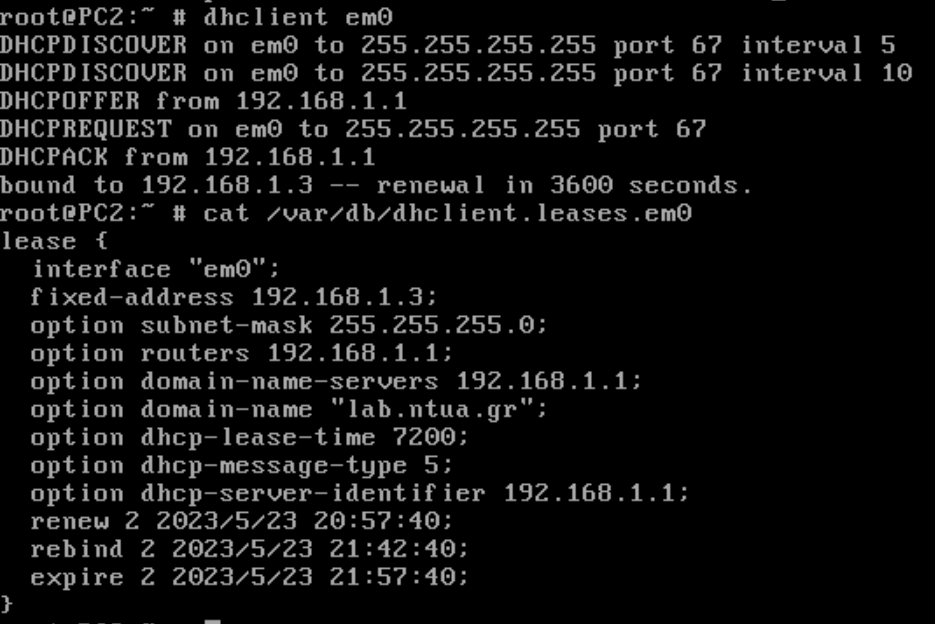
***5.38***

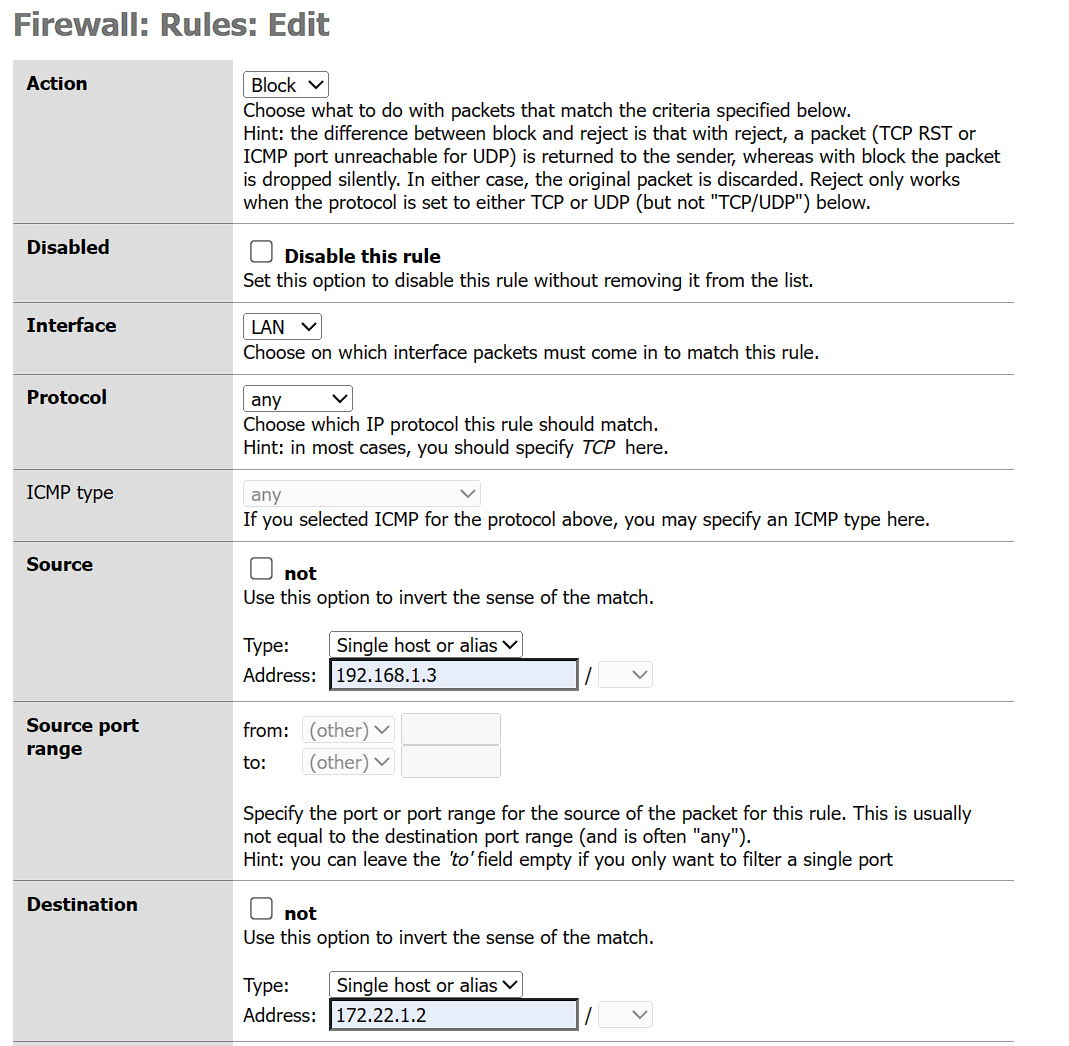
Όχι, καθώς δε μπορεί να κάνει δρομολόγηση.

***5.39***

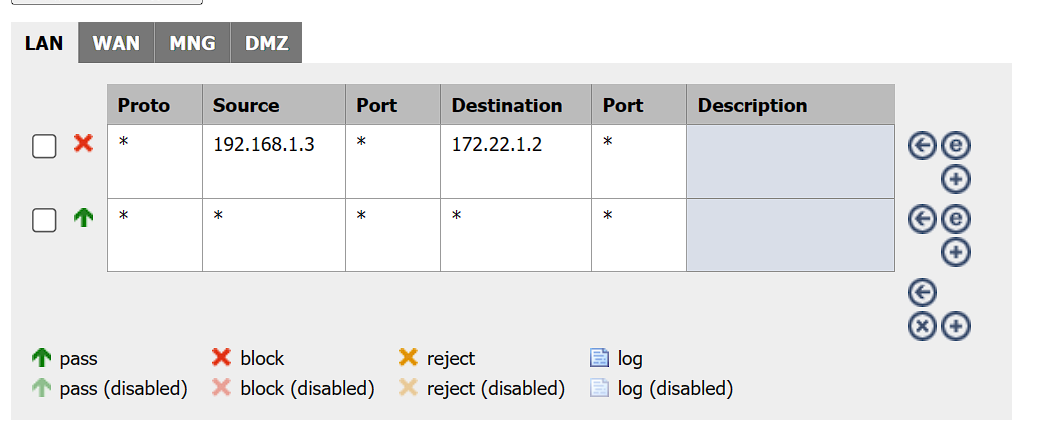
Ναι μπορούμε. Ο SRV1 στέλνει το πακέτο στο default gateway του (FW1), το οποίο και λόγω του firewall rule που βάλαμε γίνεται δεκτό. Στη συνέχεια, ο FW1 εξετάζει τον ARP πίνακά του και δεδομένου ότι το R1 δεν ανήκει στο LAN1 το προωθεί κανονικά, ενώ ο R1 απαντάει στην διεπαφή του FW1 στο WAN1.

***5.40***

IP = 192.168.1.3, Default Gateway = 192.168.1.1, DNS = 192.168.1.1.

***5.41***

***5.42***

Πρέπει να τοποθετηθεί πριν, καθώς διαφορετικά γίνεται match πρώτα ο προηγούμενος κανόνας, ο οποίος και επιτρέπει όλη την κίνηση από το LAN1 προς οπουδήποτε.

***5.43***

Όχι.

***5.44***

Ναι, καθώς απαγορεύσαμε μόνο τη διέλευση από το PC2 προς το SRV1, όχι προς όλο το DMZ.

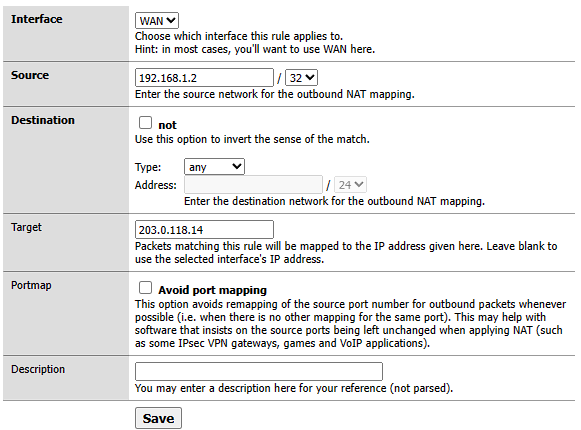
**Άσκηση 6: Τείχος προστασίας και προχωρημένο NAT**

***6.1***

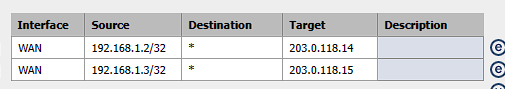
***6.2***

Εκτελούμε την αλλαγή.

***6.3***

Εκτελούμε τις αλλαγές.

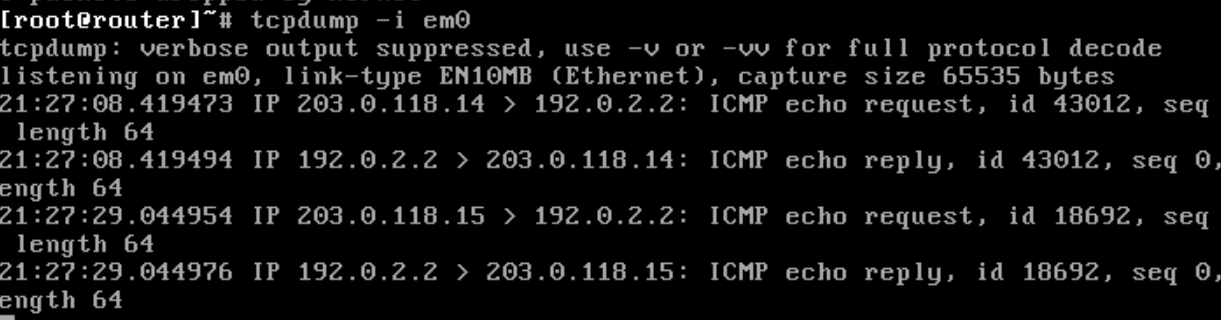
***6.4***

Εκτελούμε τις αλλαγές.

***6.5***

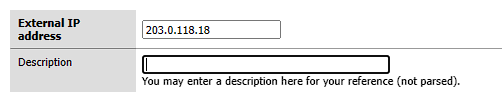
Εκτελούμε “**tcpdump -i em0**”

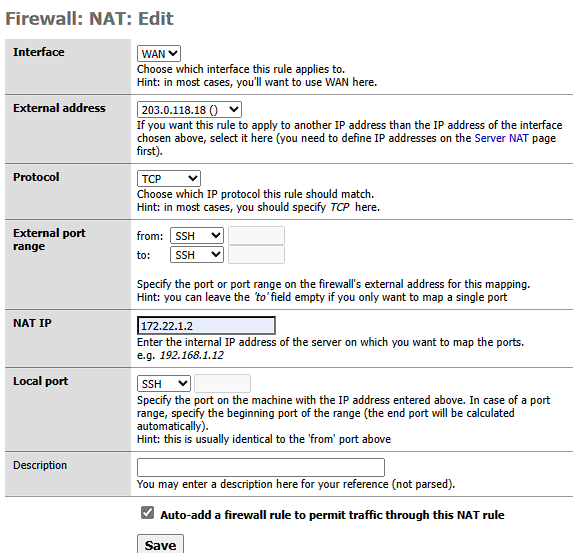
***6.6***

Μπορούμε και τα πακέτα φτάνουν με τις διευθύνσεις 203.0.118.14 και 203.0.118.15 αντίστοιχα.

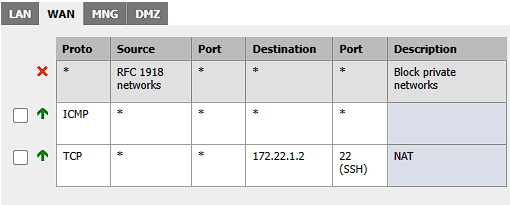
***6.7***

Αποτυγχάνει (TTL exceeded) επειδή δεν έχουμε ρύθμιση στον FW1 για inbound NAT, οπότε γίνεται αποστολή πακέτων μεταξύ των FW1 και R1 στις προεπιλεγμένες τους πύλες μεταξύ τους.

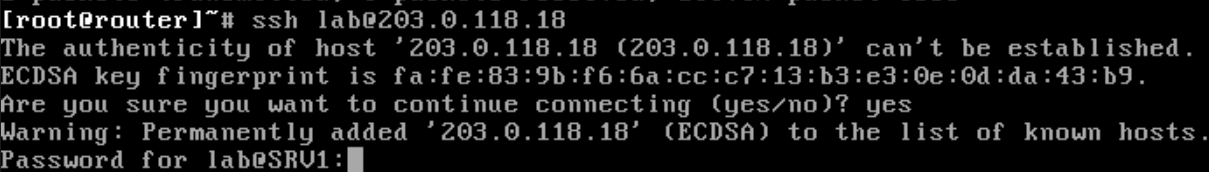
***6.8***

***6.9***

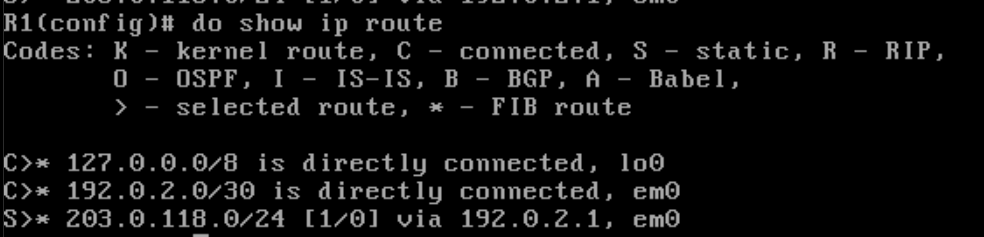
***6.10***

Βλέπουμε πως προστίθεται ο παρακάτω τρίτος κανόνας, ο οποίος επιτρέπει εισερχόμενη TCP σύνδεση προς την θύρα 22 του SRV1.

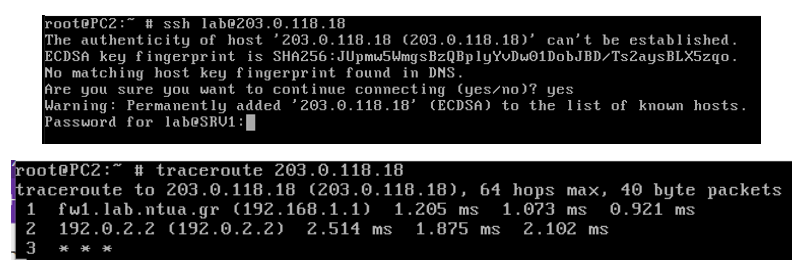
***6.11***

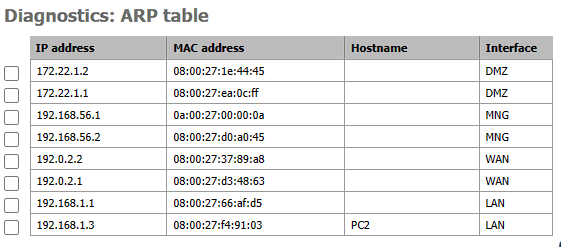
Το SRV1.

***6.12***

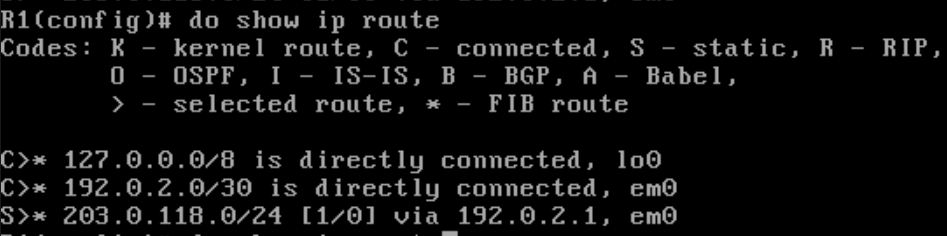
Δε μπορούμε και λαμβάνουμε ως απάντηση TTL exceeded. Βάσει του πίνακα δρομολόγησης του R1, τα πακέτα για την 203.0.118.18 δρομολογούνται στο FW1. Ωστόσο, δεν υπάρχει κατάλληλη μετάφραση όπως πριν για να φτάσουν τα πακέτα στον SRV1, καθώς επιτρέψαμε μόνο συνδέσεις στη θύρα 22 (ssh). Το FW1 τα δρομολογεί, επομένως ξανά στη δική του προκαθορισμένη πύλη, δηλαδή το R1, οπότε εμπλέκονται σε αυτό το loop μέχρι να λήξει το TTL.

***6.13***

Μπορούμε να συνδεθούμε. Για τα IP πακέτα ακολουθείται η παρακάτω διαδρομή: Το PC2 στέλνει τα IP πακέτα για το 203.0.118.18 στην προεπιλεγμένη πύλη του, δηλαδή το FW1, το οποίο με τη σειρά του, δεδομένου ότι δεν έχει εγγραφή στον ARP πίνακα για το 203.0.118.18, το προωθεί στη δική του προεπιλεγμένη πύλη, δηλαδή το R1. Ωστόσο, στον R1 προσθέσαμε στατική εγγραφή για το 203.0.118.0/24 μέσω του FW1, οπότε επαναλαμβάνεται αυτή η κίνηση μεταξύ FW1 και R1 μέχρι να μηδενιστεί το TTL.



***6.14***

Διαγράφουμε την αντιστοίχιση και δε μπορούμε πλέον να λάβουμε απάντηση στο ping. Κάνοντας “**tcpdump**” στον R1 βλέπουμε πως λαμβάνει τα Requests από τη διεύθυνση 192.168.1.2. Ωστόσο, βλέποντας τον πίνακα δρομολόγησής του, βλέπουμε πως δε μπορεί να το δρομολογήσει πίσω στον PC1.

***6.15***

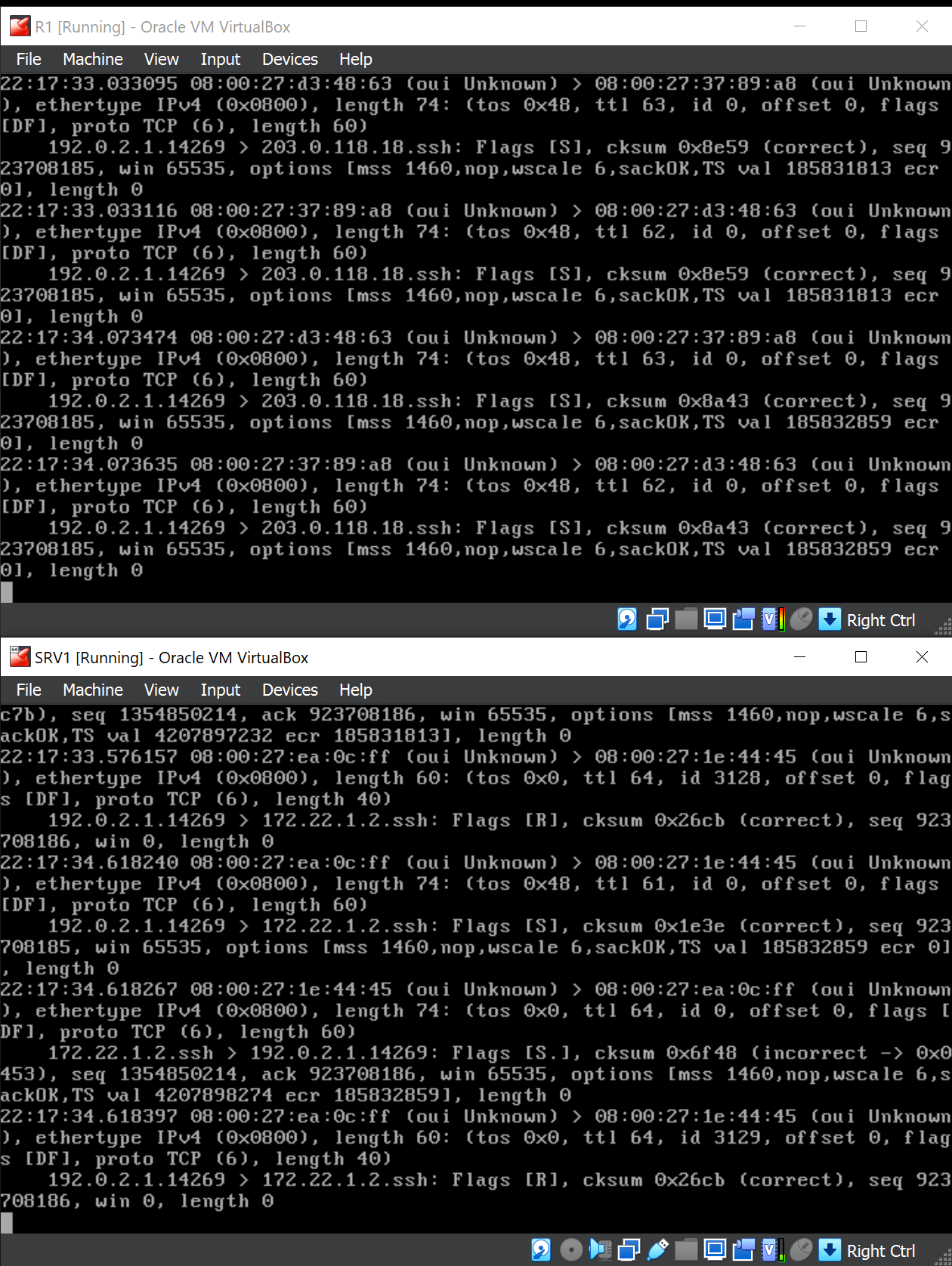
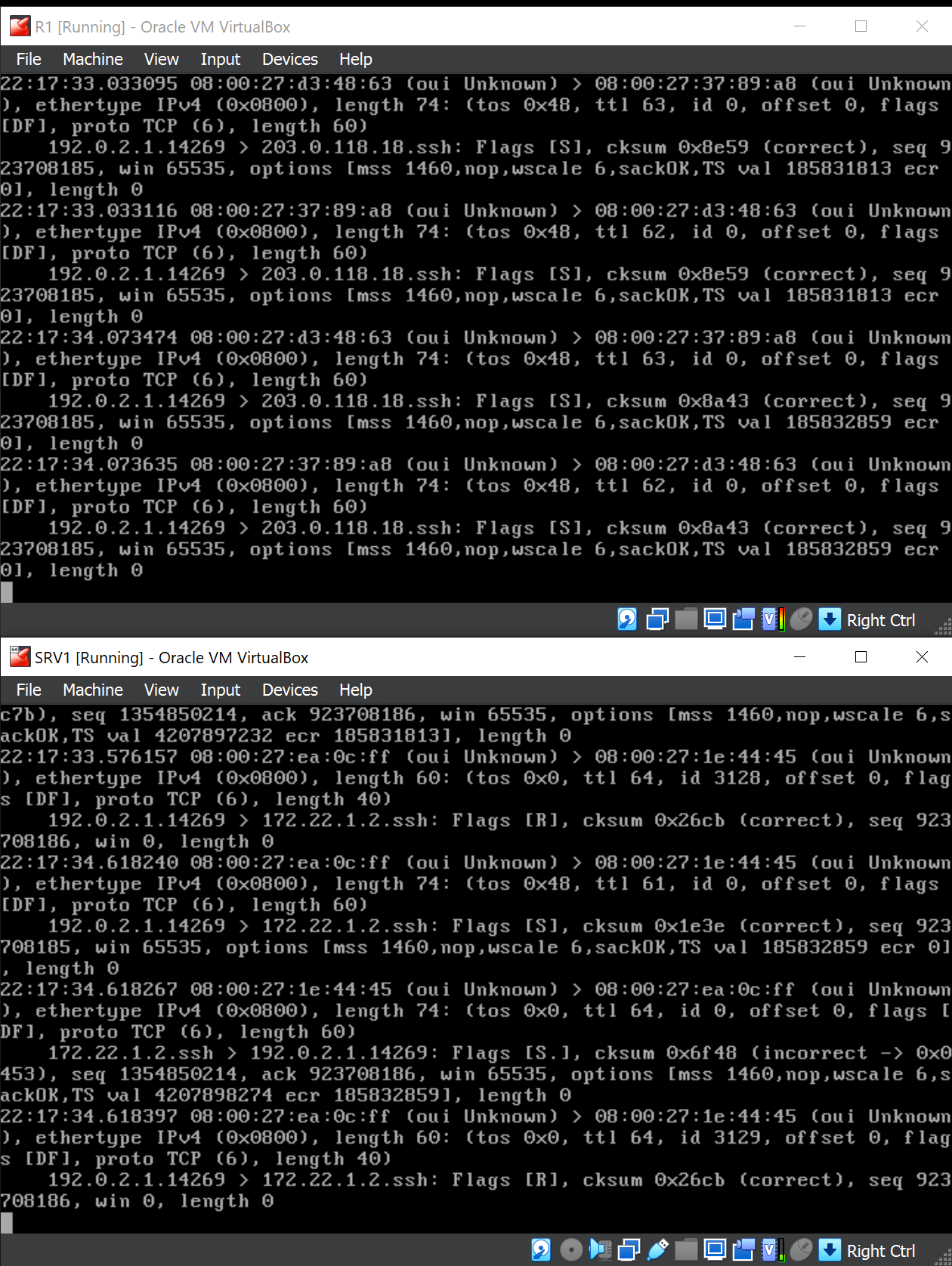
Ναι, πλέον επιτυγχάνει.

***6.16***

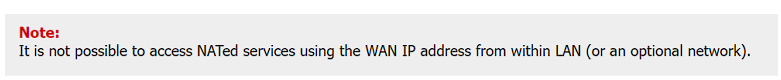
Μπορούμε να συνδεθούμε με ssh από τον R1 στον SRV1, αλλά όχι από τα PC1 και PC2.

***6.17***

Καταγράφουμε τα παρακάτω.



***6.18***

Ο παρακάτω κανόνας είναι υπεύθυνος για την παραπάνω συμπεριφορά.

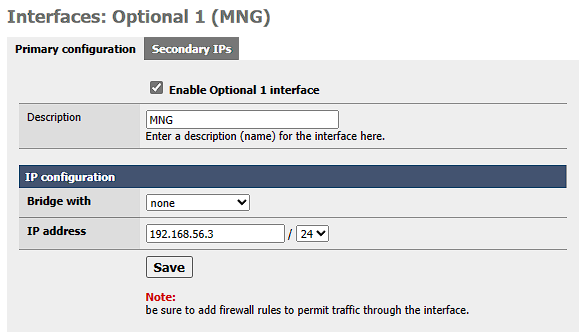
**Άσκηση 7: IPSec site-to-site Vpn**

***7.1***

Αποσυνδέουμε το καλώδιο.

***7.2***

Κάνουμε την αλλαγή και μετά πρέπει να συνδεθούμε στο “http://192.168.56.3”.



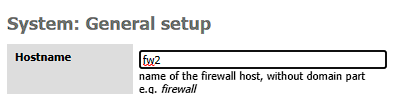
***7.3***

Επανασυνδέουμε την κάρτα.

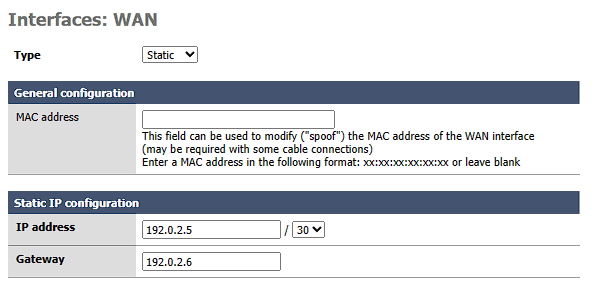
***7.4***

Ναι μπορούμε, στα “http://192.168.56.2” για το FW1 και στο “http://192.168.56.3” για το FW2.

***7.5***

Κάνουμε την αλλαγή.

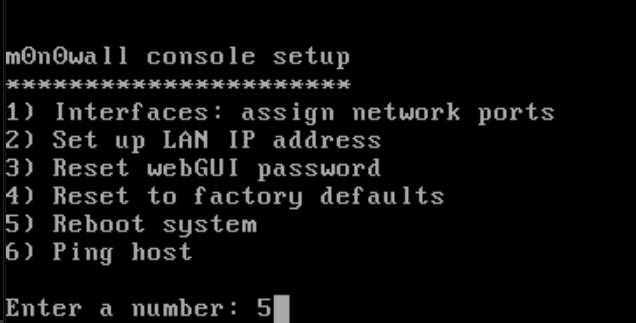
***7.6***

Κάνουμε τις αλλαγές.

***7.7***

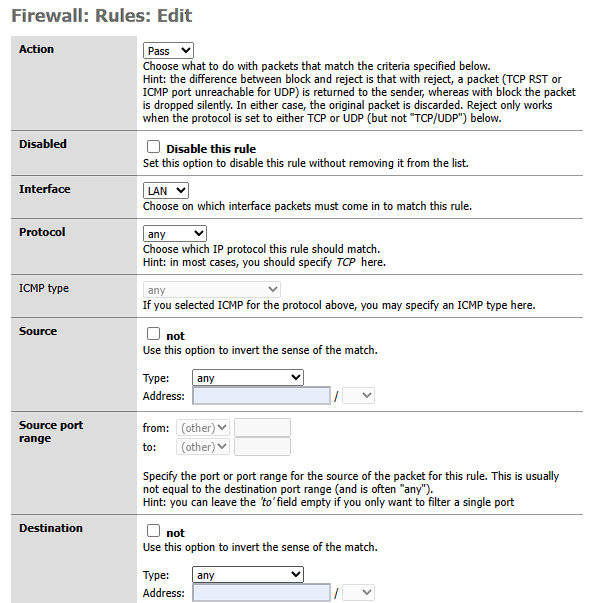
Κάνουμε την αλλαγή.

***7.8***

Κάνουμε reboot το FW2. (Εκτελούμε “**5**”)

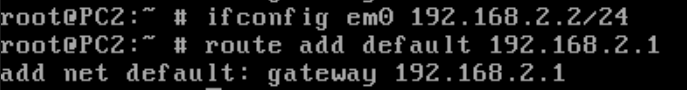
***7.9***

Προσθέτουμε τον κανόνα.



***7.10***

Όμοια με πριν:

***7.11***

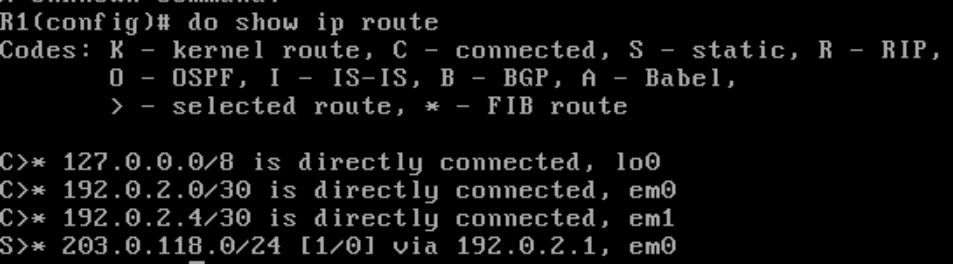
***7.12***

Ναι.

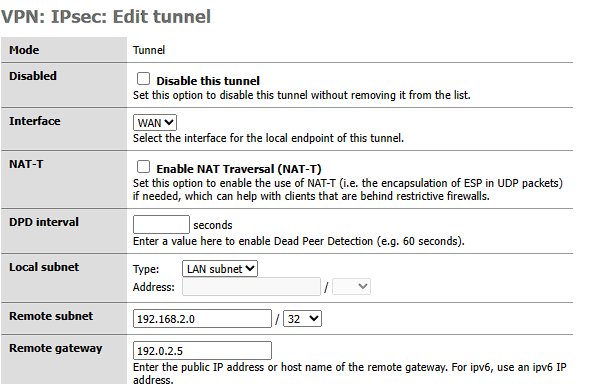
***7.13***

Ναι.

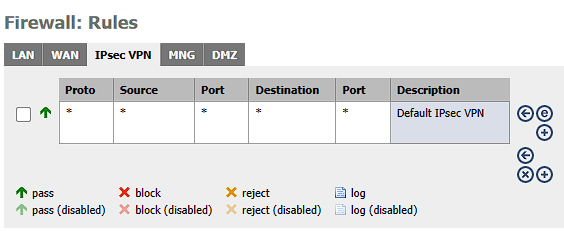
***7.14***

Η επικοινωνία αμφίδρομα είναι αδύνατη, καθώς ο R1 δε μπορεί να δρομολογήσει τα πακέτα. Παρουσιάζουμε τον πίνακα δρομολόγησής:

***7.15***



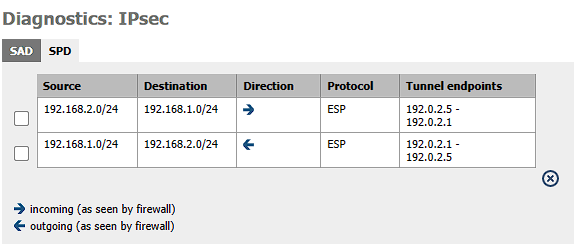
***7.16***

Βλέπουμε τον παρακάτω κανόνα.

***7.17***

Όχι.

***7.18***

Ναι.

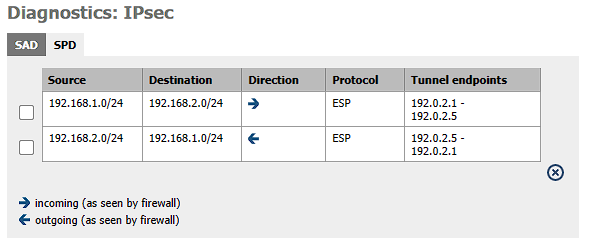
***7.19***

Κάνουμε τα ζητούμενα.

***7.20***

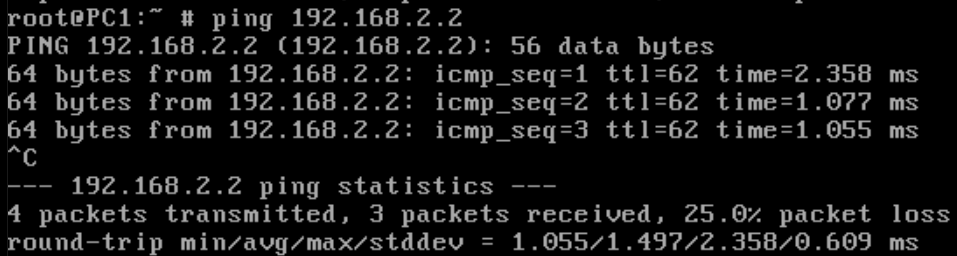
Όχι.

***7.21***

Ναι.

***7.22***

Ναι.



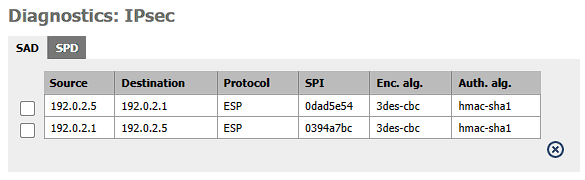
***7.23***

Ναι.

***7.24***

Ναι.

***7.25***

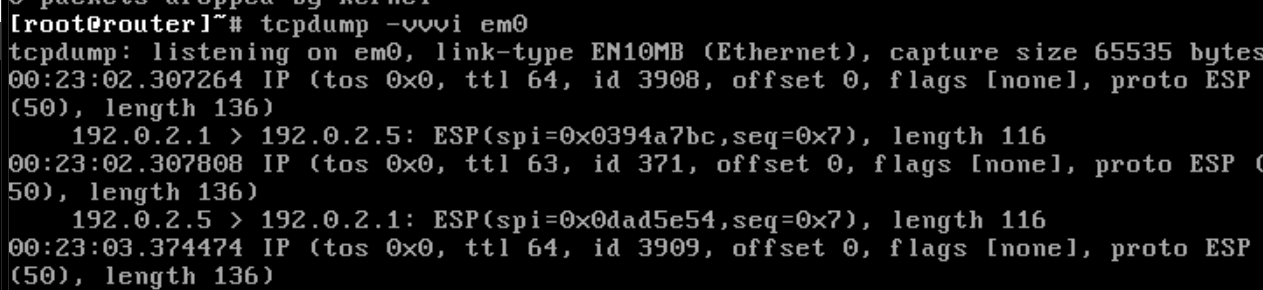
Ναι.

***7.26***

Εκτελούμε “**tcpdump -vvvi em0**” στον R1.

***7.27***

Όχι.



***7.28***

Εμφανίζονται πακέτα ESP. Το παραπάνω στιγμιότυπο είναι από το Ping του PC1 προς το PC2 και βλέπουμε πως εμφανίζεται ως διεύθυνση αποστολέα η 192.0.2.1 (διεπαφή WAN1 του FW1) και ως παραλήπτη η 192.0.2.5 (διεπαφή WAN2 του FW2).

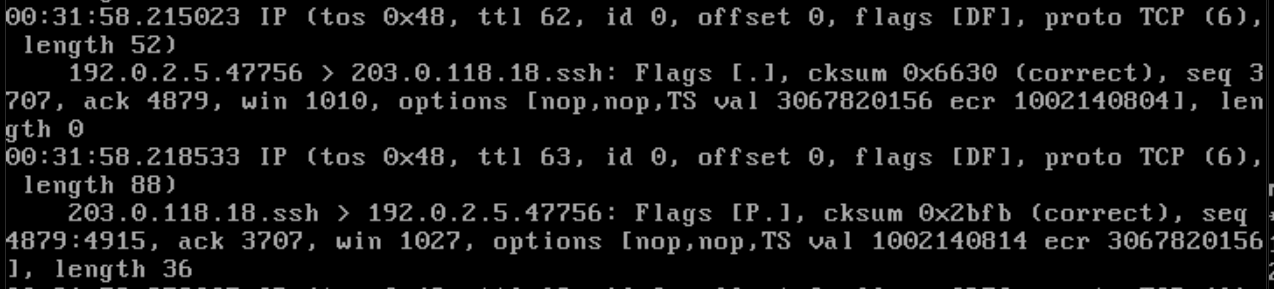
***7.29***

Δε βλέπουμε κάποια σχετική πληροφορία.

***7.30***

Ναι μπορούμε, ενώ πριν δεν μπορούσαμε.

***7.31***

Παρατηρούμε πακέτα τύπου TCP με πηγή την 192.0.2.5:47756 και προορισμό την 203.0.118.18:22 (SSH) και αντιστρόφως.

***7.32***

Είναι μεν κρυπτογραφημένα, αλλά όχι με το IPsec, αλλά με το SSH.