

Εργαστήριο Δικτύων Υπολογιστών

Εργαστηριακή Άσκηση 9 Δυναμική δρομολόγηση BGP

Όνοματεπώνυμο: Νικόλαος Παγώνας, el18175	Όνομα PC: nick-ubuntu
Ομάδα: 1 (Τρίτη 10:45)	Ημερομηνία Εξέτασης: Τρίτη 10/05/2022

Άσκηση 1: Εισαγωγή στο BGP

1

Εκτελούμε `service frr stop`.

2

Εκτελούμε `touch /usr/local/etc/frr/bgpd.conf`.

3

Εκτελούμε `chown frr:frr /usr/local/etc/frr/bgpd.conf`.

4

Αλλάζουμε τη ζητούμενη γραμμή σε `frr_daemons="zebra staticd ripd ospfd bgpd`.

5

Εκτελούμε `service frr start`.

1.1

Εκτελούμε:

```
### PC1 ###
```

```
vysh
```

```
configure terminal
```

```
hostname PC1
```

```
interface em0
ip address 192.168.1.2/24

ip route 0.0.0.0/0 192.168.1.1
```

PC2

```
vttysh
configure terminal
```

```
hostname PC2
```

```
interface em0
ip address 192.168.2.2/24

ip route 0.0.0.0/0 192.168.2.1
```

1.2

Εκτελούμε:

R1

```
cli
configure terminal
```

```
hostname R1
```

```
interface em0
ip address 192.168.1.1/24
```

```
interface em1
ip address 10.1.1.1/30
```

R2

```
cli
configure terminal
```

```
hostname R2
```

```
interface em0
ip address 10.1.1.2/30
```

```
interface em1
ip address 10.1.1.5/30
```

R3

```
cli
configure terminal

hostname R3

interface em0
ip address 10.1.1.6/30

interface em1
ip address 192.168.2.1/24
```

1.3

Στον R1 εκτελούμε `do show ip route`. Δεν υπάρχει καμία στατική εγγραφή.

1.4

Στον R1 εκτελούμε:

```
exit      # To enter global configuration mode
router ?  # BGP is available
```

1.5

Στον R1 εκτελούμε `router bgp 65010`.

1.6

Στον R1 πατάμε το πλήκτρο "?". Εμφανίζονται 14 διαθέσιμες εντολές.

1.7

Στον R1 εκτελούμε `neighbor 10.1.1.2 remote-as 65020`.

1.8

Στον R1 εκτελούμε `network 192.168.1.0/24`.

1.9

Στον R1 εκτελούμε:

```
exit
# wait for ~1 min
do show ip route
```

Δεν έχει αλλάξει κάτι στον πίνακα δρομολόγησης του R1.

1.10

Στους R1 και R2 εκτελούμε `do show ip bgp`. Στον R1 έχει προστεθεί μία εγγραφή στον πίνακα διαδρομών BGP με δίκτυο προορισμού το 192.168.1.0, ενώ στον R2 εμφανίζεται μήνυμα λάθους "No BGP process is configured".

1.11

Στον R2 εκτελούμε `router bgp 65020`.

1.12

Στον R2 εκτελούμε:

```
neighbor 10.1.1.1 remote-as 65010  
neighbor 10.1.1.6 remote-as 65030
```

1.13

Στον R2 εκτελούμε `exit` και περιμένουμε περίπου ένα λεπτό. Ύστερα στους R1 και R2 εκτελούμε `do show ip bgp`. Στον R1 δεν έχει αλλάξει κάτι, το οποίο είναι λογικό, αφού ο R2 δεν διαφημίζει κάποιο δίκτυο, ενώ στον R2 προστέθηκε μία εγγραφή με δίκτυο προορισμού το 192.168.1.0/24, την οποία ο R2 έμαθε από τον R1.

1.14

Στον R3 εκτελούμε `do show ip route`. Δεν υπάρχει διαδρομή για το 192.168.1.0/24.

1.15

Στον R3 εκτελούμε `router bgp 65030`.

1.16

Στον R3 εκτελούμε `neighbor 10.1.1.5 remote-as 65020`.

1.17

Στον R3 εκτελούμε `network 192.168.2.0/24`.

1.18

Στον R3 εκτελούμε `exit` και περιμένουμε περίπου ένα λεπτό. Ύστερα στους R1,2,3 εκτελούμε `do show ip bgp`. Πλέον και στους τρεις δρομολογητές υπάρχουν εγγραφές για τους προορισμούς 192.168.1.0/24 και 192.168.2.0/24.

1.19

Στον R2 εκτελούμε `do show ip route`. Οι εγγραφές BGP ξεχωρίζουν από το γράμμα "B" στην αρχή της γραμμής.

1.20

Δηλώνονται με το σύμβολο "*" στην αρχή της γραμμής.

1.21

Είναι 20, όπως αναγράφεται και στις εγγραφές του πίνακα δρομολόγησης, στον πρώτο αριθμό μέσα σε αγκύλες.

1.22

Στον R1 εκτελούμε `do show ip route bgp`. Βλέπουμε μία εγγραφή BGP.

1.23

Στον R1 εκτελούμε `do show ip bgp`. Βλέπουμε δύο εγγραφές, ενώ σε σχέση με τον πίνακα δρομολόγησης εμφανίζεται η επιπλέον πληροφορία Local Preference, Weight, και Path.

1.24

NETWORK	NEXT_HOP	WEIGHT	AS_PATH
192.168.1.0/24	0.0.0.0	32768	i
192.168.2.0/24	10.1.1.2	0	65020 65030 i

1.25

Στους δρομολογητές Quagga/FRR οι διαδρομές που πηγάζουν από τον δρομολογητή έχουν προκαθορισμένη τιμή WEIGHT ίση με 32768, ενώ όλες οι άλλες έχουν βάρος 0. Επομένως, η διαδρομή για το δίκτυο 192.168.1.0/24 πηγάζει από τον R1 και έχει βάρος 32768, ενώ η διαδρομή για το δίκτυο 192.168.2.0/24 δεν πηγάζει από τον R1 και έχει βάρος 0.

1.26

Παριστάνει τον τύπο πηγής ORIGIN (IGP), αφού οι διαδρομές έγιναν γνωστές μέσω της εντολής `network` στο αντίστοιχο AS.

1.27

Στον R1 εκτελούμε:

```
exit
exit
netstat -rn
```

Υπάρχει μία δυναμική εγγραφή:

Destination	Gateway	Flags	Netif
192.168.2.0/24	10.1.1.2	UG1	em1

Αυτό φαίνεται από το πεδίο Flags (UG1) και συγκεκριμένα από τη σημαία:

"1" (Protocol specific routing flag #1)

1.28

Στο PC1 εκτελούμε `do ping 192.168.2.2`. Το ping είναι επιτυχές, άρα το PC1 επικοινωνεί με το PC2.

Άσκηση 2: Λειτουργία του BGP

2.1

Στο τέλος της πρώτης γραμμής της εξόδου της εντολής `do show ip bgp neighbors` εμφανίζεται η φράση "external link" ή "internal link" αντίστοιχα.

2.2

Στην τρίτη γραμμή εμφανίζεται `BGP state = ...`, για παράδειγμα `BGP state = Established`.

2.3

Σε νέο παράθυρο στον R1 εκτελούμε `tcpdump -vni em1` και περιμένουμε τουλάχιστον ένα λεπτό.

2.4

Παρατηρούμε μηνύματα BGP KEEPALIVE.

2.5

Χρησιμοποιεί το TCP και τη θύρα 179. Η εντολή `do show ip bgp neighbors` δείχνει μόνο την θύρα που χρησιμοποιείται (αναγράφεται ως "Local port").

2.6

Τα βλέπουμε κάθε ένα λεπτό, όπως αναγράφεται και στην εντολή `do show ip bgp neighbors: keepalive interval is 60 seconds`

2.7

Το TTL είναι 1.

2.8

Στον R2 εκτελούμε `do show ip bgp summary`. Το Router-ID του R2 είναι 10.1.1.5, όπως αναγράφεται στην έξοδο της εντολής (`BGP router identifier 10.1.1.5`). Επιλέγεται αυτή η τιμή, ως η μεγαλύτερη IP διεύθυνση, εφόσον δεν έχει οριστεί διεύθυνση IP για την loopback.

2.9

Στην έξοδο της εντολής `do show ip bgp summary` βλέπουμε ότι οι 3 εγγραφές RIB καταναλώνουν 192 bytes, οπότε μία εγγραφή RIB καταναλώνει 64 bytes στη μνήμη.

2.10

Είναι 10.1.1.1 και το βρήκαμε δίνοντας την εντολή `do show ip bgp summary` στον R1.

2.11

Στον R1 εκτελούμε:

```
interface lo0  
ip address 172.17.17.1/32
```

```
do show ip bgp summary
```

Τώρα το Router-ID του R1 είναι 172.17.17.1.

2.12

Στον R1 εκτελούμε:

```
interface lo0  
no ip addr 172.17.17.1/32
```

```
do show ip bgp summary
```

Το προηγούμενο Router-ID (10.1.1.1) επανέρχεται.

2.13

Με την εντολή:

```
bgp router-id <IPaddr>
```

όπου <IPaddr> η τιμή του επιθυμητού Router-ID.

2.14

Σε νέο παράθυρο εντολών στον R2 εκτελούμε `tcpdump -vni em1`.

2.15

Στον R3 εκτελούμε:

```
router bgp 65030  
no network 192.168.2.0/24
```

2.16

Παρατηρούμε μήνυμα BGP UPDATE.

2.17

Όχι, τόσο η παραγωγή του μηνύματος όσο και η ενημέρωση του πίνακα δρομολόγησης στον R1 έγιναν αμέσως.

2.18

Στον R3 εκτελούμε `network 192.168.2.0/24`.

2.19

Ναι, υπήρξε καθυστέρηση.

2.20

Στον R1 εκτελούμε `do show ip bgp neighbors`. Παρατηρούμε ότι στάλθηκε ένα παραπάνω μήνυμα BGP UPDATE σε σχέση με πριν, με σκοπό την ενημέρωση για το νέο δίκτυο 192.168.2.0/24 που διαφημίζεται.

2.21

Έγινε με μήνυμα BGP UPDATE.

2.22

Χαρακτηριστικά:

```
ORIGIN: IGP
AS_PATH: 65020 65030
NEXT_HOP: 10.1.1.2
UPDATED ROUTES: 192.168.2.0/24
```

Άσκηση 3: Χαρακτηριστικά διαδρομών BGP

3.1

Εκτελούμε:

```
### R1 ###
```

```
interface em2
ip address 10.1.1.9/30
```

```
### R3 ###
```

```
interface em2
ip address 10.1.1.10/30
```


3.2

Στον PC1 εκτελούμε `do traceroute 192.168.2.2`. Βλέπουμε ότι ακολουθείται η διαδρομή $PC1 \rightarrow R1 \rightarrow R2 \rightarrow R3 \rightarrow PC2$.

3.3

Στον R1 εκτελούμε:

```
interface lo0
ip address 172.17.17.1/32
```

3.4

Στον R2 εκτελούμε:

```
interface lo0
ip address 172.17.17.2/32
```

3.5

Στον R3 εκτελούμε:

```
interface lo0
ip address 172.17.17.3/32
```

3.6

Εκτελούμε:

```
### R1 ###
```

```
router bgp 65010
network 172.17.17.1/32
```

```
### R2 ###
```

```
router bgp 65020
network 172.17.17.2/32
```

```
### R3 ###
```

```
router bgp 65030
network 172.17.17.3/32
```

3.7

Στον R1 εκτελούμε `do show ip bgp neighbors`. Βλέπουμε ότι ο R1 έχει γείτονα BGP τον R2 (10.1.1.2).

3.8

Στον R1 εκτελούμε `do show ip bgp`. Έχουμε:

NETWORK	NEXT_HOP
172.17.17.1/32	0.0.0.0
172.17.17.2/32	10.1.1.2
172.17.17.3/32	10.1.1.2
192.168.1.0	0.0.0.0
192.168.2.0	10.1.1.2

3.9

Στον R2 εκτελούμε `do show ip bgp neighbors`. Βλέπουμε ότι ο R2 έχει γείτονες τους R1 (10.1.1.1) και R3 (10.1.1.6).

3.10

Στον R2 εκτελούμε `do show ip bgp`. Έχουμε:

NETWORK	NEXT_HOP
172.17.17.1/32	10.1.1.1
172.17.17.2/32	0.0.0.0
172.17.17.3/32	10.1.1.6
192.168.1.0/24	10.1.1.1
192.168.2.0/24	10.1.1.6

3.11

Στον R3 εκτελούμε `do show ip bgp neighbors`. Βλέπουμε ότι ο R3 έχει γείτονα τον R2 (10.1.1.5).

3.12

Στον R3 εκτελούμε `do show ip bgp`. Έχουμε:

NETWORK	NEXT_HOP
172.17.17.1/32	10.1.1.5
172.17.17.2/32	10.1.1.5
172.17.17.3/32	0.0.0.0
192.168.1.0/24	10.1.1.5
192.168.2.0/24	0.0.0.0

3.13

Στον R1 σε νέο παράθυρο εκτελούμε `tcpdump -vni em2`.

3.14

Στον R1 εκτελούμε `neighbor 10.1.1.10 remote-as 65030`.

3.15

Στους R1 και R3 εκτελούμε `do show ip bgp neighbors`. Ο γείτονας R3 έχει προστεθεί μόνο στον R1, ενώ στον R3 δεν έχει αλλάξει κάτι.

3.16

Στον R1 εκτελούμε `do show ip bgp/route`. Η διαδρομή δεν είναι διαθέσιμη.

3.17

Στον R1 εκτελούμε `do show ip bgp neighbor`. Η σύνοδος βρίσκεται σε κατάσταση `Active`.

3.18

Στον R1 εκτελούμε `do show ip bgp summary`. Στην έξοδο της εντολής εμφανίζεται η εγγραφή:

Neighbor	AS	Up/Down	State
10.1.1.10	65030	never	Active

3.19

Βλέπουμε μηνύματα BGP OPEN.

3.20

Επαναλαμβάνεται κάθε δύο λεπτά, και ο R3 απαντά με πακέτο τερματισμού της σύνδεσης TCP (σημαία FIN ενεργοποιημένη).

3.21

Σταματάμε την καταγραφή. Μέχρι να εκτελέσουμε τα παραπάνω βήματα έγιναν προσπάθειες για να εγκατασταθεί σύνδεση TCP από τον R1, οι οποίες πετύχαιναν μόνο προσωρινά, καθώς ο R3 έστελνε αμέσως πακέτο για τερματισμό της σύνδεσης.

3.22

Στον R1 εκτελούμε `tcpdump -vni em2`.

3.23

Στον R3 εκτελούμε `neighbor 10.1.1.9 remote-as 65010`.

3.24

Στον R1 εκτελούμε `do show ip bgp neighbors`. Η σύνοδος βρίσκεται σε κατάσταση `Established`.

3.25

Στον R1 εκτελούμε `do show ip bgp`. Τώρα η διαδρομή είναι διαθέσιμη.

3.26

Στον R3 εκτελούμε `do show ip bgp`. Προστέθηκαν οι διαδρομές (επισημαίνονται με +++ στην αρχή της γραμμής):

	NETWORK	NEXT_HOP
+++	172.17.17.1/32	10.1.1.9
	172.17.17.1/32	10.1.1.5
+++	172.17.17.2/32	10.1.1.9
	172.17.17.2/32	10.1.1.5
	172.17.17.3/32	0.0.0.0
+++	192.168.1.0/24	10.1.1.9
	192.168.1.0/24	10.1.1.5
	192.168.2.0/24	0.0.0.0

3.27

Στο PC1 εκτελούμε `do traceroute 192.168.2.2`. Βλέπουμε ότι τώρα το PC1 επικοινωνεί με το PC2 μέσω της διαδρομής PC1 → R1 → R3 → PC2.

3.28

Σταματάμε την καταγραφή. Αυτή τη φορά ο R3 στέλνει μήνυμα BGP OPEN, στο οποίο ο R1 απαντάει πάλι με BGP OPEN, ενώ πριν ο R3 απλά τερμάτιζε την σύνδεση TCP.

3.29

Παρατηρήσαμε μηνύματα BGP KEEPALIVE.

3.30

```
NETWORKS: 172.17.17.1/32, 192.168.1.0/24
AS_PATH: 65010
```

```
NETWORKS: 172.17.17.2/32
AS_PATH: 65010 65020
```

```
NETWORKS: 172.17.17.3/32, 192.168.2.0/24
AS_PATH: 65010 65020 65030
```

3.31

Στον R3 εκτελούμε `do show ip bgp`. Αγνοήθηκαν οι διαδρομές προς τα δίκτυα 172.17.17.3/32 και 192.168.2.0/24, αφού αυτά τα δίκτυα ανήκουν στο τοπικό αυτόνομο σύστημα AS 65030 (το οποίο αντιλαμβάνεται ο R3 επειδή βλέπει το τοπικό αυτόνομο σύστημα AS 65030 στο AS_PATH).

3.32

Στον R1 εκτελούμε `do show ip bgp 172.17.17.2/32`. Υπάρχουν 2 διαδρομές προς τον προορισμό 172.17.17.2/32, με καλύτερη αυτή που έχει επόμενο βήμα τη διεύθυνση 10.1.1.2, δηλαδή τον R2.

3.33

Έχουμε:

Path	NEXT_HOP	ORIGIN	AS_PATH	Local Preference
#1	10.1.1.10	IGP	65030 65020	100
#2	10.1.1.2	IGP	65020	100

3.34

Ξεκινάμε από τα ισχυρότερα κριτήρια και πηγαίνουμε προς τα πιο αδύναμα, μέχρι να βρούμε κάποιο κριτήριο στο οποίο η μία διαδρομή να υπερισχύει της άλλης, οπότε θα είναι και η βέλτιστη. Τα κριτήρια είναι:

- Η διαδρομή με το υψηλότερο βάρος. Και οι δύο διαδρομές έχουν βάρος 0, καθώς δεν πηγάζουν από τον R1.
- Η διαδρομή με την υψηλότερη τιμή Local Preference. Και οι δύο διαδρομές έχουν τιμή 100.
- Η διαδρομή που ορίζεται τοπικά στο AS σε σχέση με μία που έγινε γνωστή μέσω γείτονα eBGP. Και οι δύο διαδρομές έγιναν γνωστές μέσω γείτονα BGP.
- Η διαδρομή με μικρότερο μήκος AS_PATH. Η διαδρομή #2 έχει μικρότερο μήκος AS_PATH (ίσο με 1) από την διαδρομή #1 (ίσο με 2), οπότε τελειώσαμε.

Επομένως το κριτήριο για την επιλογή είναι το μήκος του AS_PATH.

3.35

Στον R1 εκτελούμε `tcpdump -vni em2 "tcp && src 10.1.1.10 && port 179"`.

3.36

Στον R3 εκτελούμε `tcpdump -vni em0 "tcp && src 10.1.1.5 && port 179"`.

3.37

Στον R2 εκτελούμε `no network 172.17.17.2/32`.

3.38

Παράχθηκε μήνυμα BGP UPDATE με την πληροφορία:

Withdrawn routes: 5 bytes

3.39

Στον R2 εκτελούμε `network 172.17.17.2/32`.

3.40

Είναι:

```
ORIGIN: IGP
AS_PATH: 65020
NEXT_HOP: 10.1.1.5
```

3.41

Στον R2 εκτελούμε `ip route 5.5.5.0/24 100`.

3.42

Στον R2 εκτελούμε:

```
router bgp 65020
 redistribute static
```

3.43

Είναι ORIGIN: Incomplete.

3.44

Στους R1,2,3 εκτελούμε `do show ip bgp`. Η πληροφορία για τιμή ORIGIN: Incomplete δηλώνεται με το σύμβολο "?" στο τέλος της γραμμής.

Άσκηση 4: Εφαρμογή πολιτικών στο BGP

4.1

Στον R1 εκτελούμε:

```
do show ip bgp 192.168.2.0/24
```

Έχουμε:

NEXT_HOP	METRIC	LOCPRF	AS_PATH	ORIGIN
10.1.1.10	0	100	65030	IGP
10.1.1.2		100	65020 65030	IGP

4.2

Στον R3 εκτελούμε:

```
do show ip bgp 192.168.1.0/24
```

Έχουμε:

NEXT_HOP	METRIC	LOCPRF	AS_PATH	ORIGIN
10.1.1.9	0	100	65010	IGP
10.1.1.5		100	65020 65010	IGP

4.3

Στον R2 εκτελούμε:

```
do show ip bgp 192.168.1.0/24
```

```
do show ip bgp 192.168.2.0/24
```

Έχουμε:

```
### 192.168.1.0/24 ###
```

NEXT_HOP	METRIC	LOCPRF	AS_PATH	ORIGIN
10.1.1.6		100	65030 65010	IGP
10.1.1.1	0	100	65010	IGP

```
### 192.168.2.0/24 ###
```

NEXT_HOP	METRIC	LOCPRF	AS_PATH	ORIGIN
10.1.1.1		100	65010 65030	IGP
10.1.1.6	0	100	65030	IGP

4.4

Με την εντολή `do show ip bgp neighbors 10.1.1.10 advertised-routes` στον R1. Έχουμε:

Network	Next Hop	Metric	Weight	Path
*> 5.5.5.0/24	10.1.1.9		0	65020 ?
*> 172.17.17.1/32	10.1.1.9	0	32768	i
*> 172.17.17.2/32	10.1.1.9		0	65020 i
*> 192.168.1.0/24	10.1.1.9	0	32768	i

4.5

Με την εντολή `do show ip bgp neighbors 10.1.1.10 routes` στον R1. Έχουμε:

Network	Next Hop	Metric	Weight	Path
* 5.5.5.0/24	10.1.1.10		0	65030 65020 ?
* 172.17.17.2/32	10.1.1.10		0	65030 65020 i
*> 172.17.17.3/32	10.1.1.10	0	0	65030 i
*> 192.168.2.0/24	10.1.1.10	0	0	65030 i

4.6

Στον R1 εκτελούμε:

```
ip prefix-list geitones_in deny 192.168.2.0/24
```

4.7

Στον R1 εκτελούμε:

```
ip prefix-list geitones_in permit any
```

4.8

Στον R1 εκτελούμε:

```
router bgp 65010  
neighbor 10.1.1.10 prefix-list geitones_in in
```

4.9

Στον R1 εκτελούμε `do show ip bgp 192.168.2.0/24`. Δεν παρατηρούμε κάποια αλλαγή.

4.10

Στον R1 εκτελούμε `do clear ip bgp 10.1.1.10`. Εάν δεν χρησιμοποιούσαμε το `do`, θα έπρεπε να είχαμε εκτελέσει `exit` για να εισέλθουμε σε `global configuration mode`, και ύστερα ξανά `exit` για να εισέλθουμε σε `privileged exec mode`.

4.11

Στον R1 εκτελούμε `do show ip bgp neighbors 10.1.1.10 routes`. Έχει διαγραφεί η διαδρομή προς το δίκτυο 192.168.2.0/24.

4.12

Στον R1 εκτελούμε `do show ip bgp neighbors 10.1.1.10 advertised-routes`. Έχει προστεθεί η εγγραφή:

Network	Next Hop	Metric	Weight	Path
*> 192.168.2.0	10.1.1.9		0	65020 65030 i

4.13

Στον R1 εκτελούμε `do show ip bgp`. Πλέον υπάρχει μόνο μία διαδρομή για το δίκτυο 192.168.2.0/24 μέσω του R2.

4.14

Στον R2 εκτελούμε `do show ip bgp`. Πλέον υπάρχει μόνο μία διαδρομή για το δίκτυο 192.168.2.0/24 μέσω του R3.

4.15

Στον PC1 εκτελούμε `ping -R 192.168.2.2`. Ακολουθείται η διαδρομή:

```
192.168.1.2 (PC1)      # ICMP request starts
10.1.1.1 (R1)
10.1.1.5 (R2)
192.168.2.1 (R3)
192.168.2.2 (PC2)      # ICMP request is received, ICMP reply starts
10.1.1.10 (R3)
192.168.1.1 (R1)
192.168.1.2 (PC1)      # ICMP reply is received
```

4.16

Δεν την επηρεάζει, αφού ο R1 συνεχίζει να διαφημίζει το δίκτυο 192.168.1.0/24 στον R3.

4.17

Στον R1 εκτελούμε `ip prefix-list geitones_out deny 192.168.1.0/24`.

4.18

Στον R1 εκτελούμε `ip prefix-list geitones_out permit any`.

4.19

Στον R1 εκτελούμε:

```
router bgp 65010
neighbor 10.1.1.10 prefix-list geitones_out out
```

4.20

Στον R1 εκτελούμε `do clear ip bgp 10.1.1.10`.

4.21

Στον R1 εκτελούμε `do show ip bgp neighbor 10.1.1.10 advertised-routes`. Έχει διαγραφεί η διαδρομή προς το δίκτυο 192.168.1.0/24.

4.22

Στον R1 εκτελούμε `do show ip bgp neighbor 10.1.1.10 routes`. Δεν παρατηρούμε κάποια αλλαγή.

4.23

Στον R3 εκτελούμε `do show ip bgp`. Πλέον προς το 192.168.1.0/24 υπάρχει μόνο μία διαδρομή που έχει Next Hop τη διεύθυνση 10.1.1.5, δηλαδή τον R2.

4.24

Στον R2 εκτελούμε `do show ip bgp`. Πλέον προς το 192.168.2.0/24 υπάρχει μόνο μία διαδρομή που έχει Next Hop τη διεύθυνση 10.1.1.1, δηλαδή τον R1.

4.25

Στον PC1 εκτελούμε `ping -R 192.168.2.2`. Ακολουθείται η διαδρομή:

```
192.168.1.2 (PC1) # ICMP request starts
10.1.1.1 (R1)
10.1.1.5 (R2)
192.168.2.1 (R3)
192.168.2.2 (PC2) # ICMP request is received, ICMP reply starts
10.1.1.6 (R3)
10.1.1.2 (R2)
192.168.1.1 (R1)
192.168.1.2 (PC1) # ICMP reply is received
```

4.26

Στον R1 εκτελούμε:

```
no neighbor 10.1.1.10 prefix-list geitones_in in
no neighbor 10.1.1.10 prefix-list geitones_out out
do clear ip bgp 10.1.1.10
```

Άσκηση 5: iBGP

5.1

Στον R4 εκτελούμε:

```
cli
configure terminal

hostname R4

interface em0
ip address 192.168.0.2/24

interface em1
ip address 10.1.1.13/30
```

5.2

Στον R4 εκτελούμε:

```
interface lo0
ip address 172.17.17.4/32
```

5.3

Στον R1 εκτελούμε:

```
interface em3
ip address 192.168.0.1/24
```

5.4

Στον R3 εκτελούμε:

```
interface em3
ip address 10.1.1.14/30
```

5.5

Στον R4 εκτελούμε:

```
router bgp 65010
```

5.6

Στον R4 εκτελούμε:

```
neighbor 192.168.0.1/24 remote-as 65010
network 172.17.17.4/32
```

5.7

Στον R1 εκτελούμε:

```
neighbor 192.168.0.2/24 remote-as 65010
```

5.8

Στον R1 εκτελούμε `do show ip bgp neighbors 192.168.0.2`. Η σύνοδος είναι internal, αφού στην πρώτη γραμμή αναγράφεται "internal link".

5.9

Στον R4 εκτελούμε `do show ip bgp neighbors 192.168.0.1 routes` και βλέπουμε ότι οι διαδρομές που έμαθε από τον R1 και συμπεριέλαβε στην RIB είναι (αναγράφεται ακριβώς δίπλα και το NEXT_HOP):

Network	Next Hop
5.5.5.0/24	10.1.1.2
172.17.17.1/32	192.168.0.1
172.17.17.2/32	10.1.1.2
172.17.17.3/32	10.1.1.10
192.168.1.0	192.168.0.1
192.168.2.0	10.1.1.10

5.10

Αντίστοιχα, στον R1 εκτελούμε `do show ip bgp neighbors 192.168.0.2 routes` και έχουμε:

Network	Next Hop
172.17.17.4/32	192.168.0.2

5.11

Τις διακρίνουμε επειδή έχουν το σύμβολο "i" στην αρχή της γραμμής (διαφορετικό από το "i" στο τέλος της γραμμής -βλ. ερώτημα 1.26-).

5.12

Ναι, έχουν τεθεί ρητά οι τιμές `Metric = 0` και `Local Preference = 100`.

5.13

Στον R4 εκτελούμε `do show ip route`. Από τα δίκτυα της ερώτησης 5.9 έχουν εισαχθεί στον πίνακα δρομολόγησης τα:

172.17.17.1/32
192.168.1.0/24

5.14

Δεν έχουν εισαχθεί τα:

5.5.5.0/24
172.17.17.2/32
172.17.17.3/32
192.168.2.0/24

Αυτό συμβαίνει επειδή τα NEXT_HOP που αφορούν τα παραπάνω δίκτυα δεν είναι προσβάσιμα, δηλαδή δεν υπάρχει εγγραφή γι' αυτά στον πίνακα δρομολόγησης του R4.

5.15

Στον R4 εκτελούμε `ip route 10.1.1.8/30 192.168.0.1`.

5.16

Στον R4 εκτελούμε `do show ip route`. Τώρα το 192.168.2.0/24 έχει τοποθετηθεί στον πίνακα δρομολόγησης του R4. Για το επόμενο βήμα αναγράφεται:

via 10.1.1.10 (recursive via 192.168.0.1)

5.17

Από τα:

5.5.5.0/24
172.17.17.2/32
172.17.17.3/32
192.168.2.0/24

δεν έχουν εισαχθεί τα:

5.5.5.0/24
172.17.17.2/32,

επειδή το NEXT_HOP τους (10.1.1.2) εξακολουθεί να μην είναι προσβάσιμο από τον R4.

5.18

Στον R1 εκτελούμε `neighbor 192.168.0.2 next-hop-self`.

5.19

Στον R4 εκτελούμε `do show ip route`. Παρατηρούμε ότι πλέον έχουν εισαχθεί και τα δίκτυα

5.5.5.0/24
172.17.17.2/32

στον πίνακα δρομολόγησης. Επίσης δεν αναγράφεται πλέον το μήνυμα (`recursive via 192.168.0.1`) ενώ ως NEXT_HOP για όλες τις διαδρομές iBGP είναι το 192.168.0.1, δηλαδή ο R1.

5.20

Στον R4 εκτελούμε `do show ip route`. Η διαχειριστική απόσταση είναι 200, γιατί αυτή τη φορά πρόκειται για Internal BGP, ενώ στην ερώτηση 1.21 επρόκειτο για External BGP (διαχειριστική απόσταση 20).

5.21

Στον R4 εκτελούμε `do ping 10.1.1.9`. Το ping είναι επιτυχές.

5.22

Στον R4 εκτελούμε `do ping 10.1.1.10`. Δεν λαμβάνουμε καμία απάντηση. Αυτό συμβαίνει διότι, ενώ το ICMP request φτάνει στον R3, δεν μπορεί να σταλεί ICMP reply πίσω στον R4, επειδή δεν υπάρχει εγγραφή στον πίνακα δρομολόγησης του R3 σχετική με τον R4 (είτε εγγραφή για το δίκτυο 192.168.0.0/24 είτε προκαθορισμένη διαδρομή)

5.23

Στον R1 εκτελούμε `network 192.168.0.0/24`.

5.24

Στον R4 εκτελούμε πάλι `do ping 10.1.1.10`. Πλέον το ping είναι επιτυχές.

5.25

Στον R1 εκτελούμε `aggregate-address 192.168.0.0/23`.

5.26

Στον R3 εκτελούμε `do show ip bgp`. Πλέον βλέπουμε 6 εγγραφές σχετικές με το 192.168.0.0/23 και τα υποδίκτυά του:

Network	Next Hop
192.168.0.0/23	10.1.1.9
192.168.0.0/23	10.1.1.5
192.168.0.0	10.1.1.5
192.168.0.0	10.1.1.9
192.168.1.0	10.1.1.9
192.168.1.0	10.1.1.5

5.27

Στον R1 εκτελούμε `aggregate-address 192.168.0.0/23 summary-only`.

5.28

Στον R3 εκτελούμε `do show ip bgp`. Πλέον βλέπουμε 2 εγγραφές σχετικές με το 192.168.0.0/23:

Network	Next Hop
192.168.0.0/23	10.1.1.9
192.168.0.0/23	10.1.1.5

5.29

Στον R1 εκτελούμε `no aggregate-address 192.168.0.0/23 summary-only`.

5.30

Στον R4, σε νέο παράθυρο, εκτελούμε `tcpdump -vni em0`.

5.31

Το TTL των πακέτων που μεταφέρουν τα μηνύματα BGP είναι ίσο με 64 και όχι 1, όπως είχαμε βρει στην ερώτηση 2.7. Αυτό συμβαίνει γιατί αυτή τη φορά έχουμε iBGP (default TTL 64), και όχι eBGP (default TTL 1).

Άσκηση 6: Περισσότερα περί πολιτικών στο BGP

6.1

Εκτελούμε:

```
### R4 ###  
neighbor 10.1.1.14 remote-as 65030
```

```
### R3 ###  
neighbor 10.1.1.13 remote-as 65010
```

6.2

Στον R4 εκτελούμε `neighbor 192.168.0.1 next-hop-self`.

6.3

Στον R1 εκτελούμε `do show ip bgp`. Υπάρχουν 3 διαδρομές προς το 192.168.2.0/24:

	Network	Next Hop
#1	192.168.2.0/24	10.1.1.14
#2	192.168.2.0/24	10.1.1.10
#3	192.168.2.0/24	10.1.1.2

Από αυτές έχει επιλεχθεί η διαδρομή #2, με NEXT_HOP τη διεύθυνση 10.1.1.10, δηλαδή τον R3.

6.4

Ελέγχουμε ένα-ένα τα κριτήρια ξεκινώντας από το σημαντικότερο και σταματάμε όταν μόνο μία διαδρομή υπερτερεί σε κάποιο:

Συν.: σημαίνει ότι η διαδρομή συνεχίζει και στο επόμενο κριτήριο.

Απ.: σημαίνει ότι η διαδρομή αποκλείεται με βάση το τρέχον κριτήριο.

—: σημαίνει ότι η διαδρομή έχει αποκλειστεί πιο πριν και δεν λαμβάνεται πλέον υπόψιν.

Επ.: σημαίνει ότι η διαδρομή επιλέγεται ως καλύτερη και η αναζήτηση σταματά.

Κριτήριο	#1	#2	#3	Αιτιολόγηση
WEIGHT	Συν.	Συν.	Συν.	Ίδιο WEIGHT για όλες
LOCAL_PREF	Συν.	Συν.	Συν.	Ίδιο LOCAL_PREF για όλες
AS_PATH	Συν.	Συν.	Απ.	Οι #1 και #2 έχουν μήκος 1, ενώ η #3 έχει μήκος 2
ORIGIN	Συν.	Συν.	—	Όλες είναι IGP (i)
MED	Συν.	Συν.	—	Δεν έχει οριστεί MED σε καμία διαδρομή
eBGP vs iBGP	Απ.	Επ.	—	#1 γνωστή μέσω γείτονα iBGP, ενώ #2 μέσω γείτονα eBGP.

Τελικά η επιλογή βασίστηκε στο ότι η διαδρομή #2 (NEXT_HOP 10.1.1.10) έγινε γνωστή μέσω γείτονα eBGP, ενώ η διαδρομή #1 μέσω γείτονα iBGP, γι' αυτό και επιλέχθηκε η διαδρομή #2.

6.5

Στον R4 εκτελούμε `do show ip bgp`. Υπάρχουν 2 διαδρομές προς το 192.168.2.0/24:

	Network	Next Hop
#1	192.168.2.0/24	10.1.1.14
#2	192.168.2.0/24	192.168.0.1

Από αυτές έχει επιλεχθεί η διαδρομή #1, με NEXT_HOP τη διεύθυνση 10.1.1.14, δηλαδή τον R3.

6.6

Αντίστοιχα με το ερώτημα 6.4, ελέγχουμε τα κριτήρια ξεκινώντας από το σημαντικότερο. Και πάλι το καθοριστικό κριτήριο είναι ότι η διαδρομή #1 (NEXT_HOP 10.1.1.14) έγινε γνωστή μέσω γείτονα eBGP, ενώ η διαδρομή #2 μέσω γείτονα iBGP, οπότε επιλέχθηκε η διαδρομή #1.

6.7

Στον R4 εκτελούμε `do show ip bgp`. Υπάρχουν 2 διαδρομές προς το 172.17.17.2/32:

	Network	Next Hop
#1	172.17.17.2/32	10.1.1.14
#2	172.17.17.2/32	192.168.0.1

Από αυτές έχει επιλεχθεί η διαδρομή #2, με NEXT_HOP τη διεύθυνση 192.168.0.1, δηλαδή τον R1.

6.8

Αντίστοιχα με τα ερωτήματα 6.4 και 6.6, ελέγχουμε τα κριτήρια ξεκινώντας από το σημαντικότερο. Αυτή τη φορά καθοριστικό κριτήριο είναι ότι η διαδρομή #2 (NEXT_HOP 192.168.0.1) έχει μικρότερο μήκος AS_PATH (ίσο με 1) από τη διαδρομή #1 (ίσο με 2), οπότε επιλέγεται η διαδρομή #2.

6.9

Στον R3 εκτελούμε `do show ip bgp`. Υπάρχουν 3 διαδρομές προς το 192.168.1.0/24:

	Network	Next Hop
#1	192.168.1.0/24	10.1.1.13
#2	192.168.1.0/24	10.1.1.5
#3	192.168.1.0/24	10.1.1.9

Από αυτές έχει επιλεχθεί η διαδρομή #3, με NEXT_HOP τη διεύθυνση 10.1.1.9, δηλαδή τον R1.

6.10

Σύμφωνα με την υπόδειξη της ερώτησης, εκτελούμε `do show ip bgp 192.168.1.0/24`. Αρχικά η διαδρομή #2 αποκλείεται επειδή έχει μεγαλύτερο μήκος AS_PATH (ίσο με 2) από τις #1 και #3 (ίσο με 1). Ύστερα, μεταξύ των διαδρομών #1 και #3 επιλέγεται η #3 επειδή είναι αρχαιότερη, όπως δείχνει και η τιμή του πεδίου Last Update.

6.11

Στον R1 εκτελούμε `do clear ip bgp 10.1.1.10` και ύστερα από λίγο εκτελούμε στον R3 `do show ip bgp 192.168.1.0/24`. Παρατηρούμε ότι πλέον έχει επιλεχθεί η διαδρομή με επόμενο βήμα τη διεύθυνση 10.1.1.13, δηλαδή τον R4.

6.12

Στον R4 εκτελούμε `do clear ip bgp 10.1.1.14` και ύστερα από λίγο εκτελούμε στον R3 `do show ip bgp 192.168.1.0/24`. Παρατηρούμε ότι έχει επιλεχθεί η διαδρομή με επόμενο βήμα τη διεύθυνση 10.1.1.9, δηλαδή τον R1.

6.13

Στον R1 εκτελούμε:

```
exit          # To enter global configuration mode

route-map set-locpref permit 10
```

6.14

Στον R1 εκτελούμε:

```
set local-preference 90

exit
```

6.15

Στον R1 εκτελούμε:

```
router bgp 65010

neighbor 10.1.1.10 route-map set-locpref in
```

6.16

Στον R1 εκτελούμε `do clear ip bgp *` και περιμένουμε περίπου ένα λεπτό. Ύστερα εκτελούμε `do show ip bgp`. Η τιμή του local-preference έχει αλλάξει στις διαδρομές:

Network	Next Hop	LocPrf
5.5.5.0/24	10.1.1.10	90
172.17.17.2/32	10.1.1.10	90
172.17.17.3/32	10.1.1.10	90
192.168.2.0/24	10.1.1.10	90

6.17

Στον R1 εκτελούμε `do show ip route`. Έχει επιλεγθεί η διαδρομή με επόμενο βήμα τη διεύθυνση 192.168.0.2, δηλαδή τον R4, επειδή έχει μεγαλύτερο `local-preference` (ίσο με 100) από τη διαδρομή με επόμενο βήμα τη διεύθυνση 10.1.1.10 (ίσο με 90), και επίσης έχει μικρότερο μήκος `AS_PATH` (ίσο με 1) από τη διαδρομή με επόμενο βήμα τη διεύθυνση 10.1.1.2 (ίσο με 2).

6.18

Στον R4 εκτελούμε `do show ip bgp`. Παρατηρούμε ότι πλέον υπάρχει μόνο μία διαδρομή προς το δίκτυο 192.168.2.0/24 με επόμενο βήμα τη διεύθυνση 10.1.1.14, δηλαδή τον R3. Επιπλέον, υπάρχει μόνο μία διαδρομή προς το δίκτυο 172.17.17.3/32 με επόμενο βήμα τη διεύθυνση 10.1.1.14, δηλαδή τον R3.

6.19

Στον R1 εκτελούμε:

```
do show ip bgp neighbors 192.168.0.2 advertised-routes
```

Εμφανίζονται διαδρομές για τα δίκτυα:

```
5.5.5.0/24
172.17.17.1/32
172.17.17.2/32
192.168.0.0/24
192.168.1.0/24
```

Παρατηρούμε ότι δεν εμφανίζονται διαδρομές για τα δίκτυα του AS 65030.

6.20

Οι αλλαγές στην RIB του R4 που παρατηρήσαμε προηγουμένως οφείλονται στο ότι ο R1 δεν διαφημίζει πλέον διαδρομές προς τα δίκτυα του AS 65030, όπως είδαμε παραπάνω. Αυτό συμβαίνει επειδή η αλλαγή του `local-preference` στις διαδρομές που διαφημίζει ο R3 στον R1 ωθεί τον R1 να προτιμά διαδρομές με επόμενο βήμα τον R4, και όχι τον R3.

6.21

Στο PC1 εκτελούμε `ping -R 192.168.2.2`. Ακολουθείται η διαδρομή:

```
192.168.1.2 (PC1)      # ICMP request starts
192.168.0.1 (R1)
10.1.1.13  (R4)
192.168.2.1 (R3)
192.168.2.2 (PC2)      # ICMP request is received, ICMP reply starts
10.1.1.10  (R3)
192.168.1.1 (R1)
192.168.1.2 (PC1)      # ICMP reply is received
```

6.22

Στον R1 εκτελούμε:

```
exit          # To enter global configuration mode

route-map set-MED permit 15
```

6.23

Στον R1 εκτελούμε:

```
set metric 1
exit
```

6.24

Στον R1 εκτελούμε:

```
router bgp 65010
neighbor 10.1.1.10 route-map set-MED out
```

6.25

Στον R1 εκτελούμε `do clear ip bgp 10.1.1.10` και ύστερα από λίγο εκτελούμε στον R3 `do show ip bgp`. Η τιμή του Metric έχει αλλάξει στις διαδρομές:

Network	Next Hop	Metric
5.5.5.0/24	10.1.1.9	1
172.17.17.1/32	10.1.1.9	1
172.17.17.2/32	10.1.1.9	1
172.17.17.4/32	10.1.1.9	1
192.168.0.0/24	10.1.1.9	1
192.168.1.0/24	10.1.1.9	1

6.26

Στον R3 εκτελούμε `do show ip bgp`. Έχει επιλεγθεί η διαδρομή με επόμενο βήμα τη διεύθυνση 10.1.1.13, δηλαδή τον R4, επειδή έχει μικρότερο μήκος AS_PATH από την διαδρομή με επόμενο βήμα τη διεύθυνση 10.1.1.5 (1 έναντι 2), και επιπλέον επειδή έχει μικρότερο MED από τη διαδρομή με επόμενο βήμα τη διεύθυνση 10.1.1.9 (0 έναντι 1). Δίνεται έμφαση στο ότι οι διαδρομές με επόμενο βήμα 10.1.1.9 και 10.1.1.13 έχουν ίδιο πρώτο βήμα AS, αλλιώς δεν θα είχε νόημα η σύγκριση του MED.

6.27

Στον PC1 εκτελούμε `ping -R 192.168.2.2`. Αυτή τη φορά ακολουθείται η διαδρομή:

```
192.168.1.2 (PC1)      # ICMP request starts
192.168.0.1 (R1)
10.1.1.13  (R4)
192.168.2.1 (R3)
192.168.2.2 (PC2)      # ICMP request is received, ICMP reply starts
10.1.1.14  (R3)
192.168.0.2 (R4)
192.168.1.1 (R1)
192.168.1.2 (PC1)      # ICMP reply is received
```

6.28

Στον R1 εκτελούμε:

```
exit #          To enter global configuration mode

route-map set-prepend permit 5
```

6.29

Στον R1 εκτελούμε:

```
set as-path prepend 65010 65010

exit
```

6.30

Στον R1 εκτελούμε:

```
router bgp 65010
neighbor 10.1.1.2 route-map set-prepend out
```

6.31

Στον R1 εκτελούμε `do clear ip bgp 10.1.1.2` και ύστερα από λίγο στον R2 εκτελούμε `do show ip bgp`. Παρατηρούμε ότι στο AS_PATH των εγγραφών που έχουν ως επόμενο βήμα τη διεύθυνση 10.1.1.1 εμφανίζεται 3 φορές το ASN 65010, εξαιτίας του route-map που εφαρμόσαμε, σύμφωνα με το οποίο ο R1 προσθέτει 2 παραπάνω φορές το 65010 στο AS_PATH των διαδρομών που διαφημίζει στον R2.

6.32

Στον R2 εκτελούμε `do show ip route bgp`. Το επόμενο βήμα σε όλες τις διαδρομές είναι η διεύθυνση 10.1.1.6, δηλαδή ο R3.

6.33

Στον R3 εκτελούμε `do show ip bgp`. Παρατηρούμε ότι έχουν διαγραφεί οι εγγραφές:

Network	Next Hop
172.17.17.1/32	10.1.1.5
172.17.17.4/32	10.1.1.5
192.168.0.0/24	10.1.1.5
192.168.1.0/24	10.1.1.5

όσες δηλαδή είχαν ως επόμενο βήμα τον R2.

6.34

Στον R4 εκτελούμε `do show ip bgp` και επιβεβαιώνουμε ότι δεν έχει αλλάξει τίποτα. Αυτό συμβαίνει διότι οι διαδρομές που διαγράφηκαν στον R3 δεν διαφημίζονταν έτσι κι αλλιώς από τον R3, αφού δεν είχαν επιλεγθεί στον πίνακα δρομολόγησης.

Άσκηση 7: Περισσότερα για το iBGP και την προκαθορισμένη διαδρομή

7.1

Στο PC1 εκτελούμε:

```
no ip route 0.0.0.0/0 192.168.1.1

router bgp 65010

neighbor 192.168.1.1 remote-as 65010
```

7.2

Στον R1 εκτελούμε `neighbor 192.168.1.2 remote-as 65010`.

7.3

Στον PC1 εκτελούμε:

```
do show ip bgp
do show ip route
```

Παρατηρούμε ότι στον πίνακα δρομολόγησης έχουν εισαχθεί μόνο τα δίκτυα:

```
172.17.17.1/32
192.168.0.0/24
192.168.1.0/24
```

δηλαδή αυτά που ανήκουν στο AS 65010. Αυτό συμβαίνει διότι οι υπόλοιποι προορισμοί έχουν ως επόμενο βήμα (NEXT_HOP) διευθύνσεις οι οποίες δεν είναι προσβάσιμες από τον PC1, δηλαδή δεν υπάρχουν στον πίνακα δρομολόγησης του.

7.4

Στον R1 εκτελούμε `neighbor 192.168.1.2 next-hop-self` και ύστερα από λίγο στο PC1 εκτελούμε `do show ip route`. Τώρα ο PC1 γνωρίζει διαδρομές προς τα δίκτυα:

```
5.5.5.0/24          # new
172.17.17.1/32
172.17.17.2/32      # new
192.168.0.0/24
192.168.1.0/24
```

δηλαδή προς όλους τους προορισμούς που έμαθε από τον R1.

7.5

Διότι ο R1 δεν διαφημίζει διαδρομές προς τα δίκτυα:

```
172.17.17.3/32, 172.17.17.4/32, 192.168.2.0/24
```

στον PC1, διότι αυτές έγιναν γνωστές από άλλο εσωτερικό συνομιλητή (τον R4), και επομένως δεν προωθούνται σε άλλους εσωτερικούς συνομιλητές.

7.6

Εκτελούμε:

```
### PC1 ###
```

```
neighbor 192.168.0.2 remote-as 65010
```

```
### R4 ###
```

```
neighbor 192.168.1.2 remote-as 65010
```

7.7

Πρέπει να εκτελέσουμε στον R4 `neighbor 192.168.1.2 next-hop-self`.

7.8

Στο PC1 εκτελούμε:

```
do ping 192.168.1.2      # Network: LAN1 (192.168.1.0/24)
do ping 192.168.2.2      # Network: LAN2 (192.168.2.0/24)
do ping 192.168.0.2      # Network: LAN3 (192.168.0.0/24)
do ping 10.1.1.2         # Network: WAN1 (10.1.1.0/30)
do ping 10.1.1.6         # Network: WAN2 (10.1.1.4/30)
do ping 10.1.1.10        # Network: WAN3 (10.1.1.8/30)
do ping 10.1.1.14        # Network: WAN5 (10.1.1.12/30)
do ping 172.17.17.1      # Network: 172.17.17.1/32
```

```
do ping 172.17.17.2          # Network: 172.17.17.2/32
do ping 172.17.17.3          # Network: 172.17.17.3/32
do ping 172.17.17.4          # Network: 172.17.17.4/32
```

Τα ping προς τα LAN1, 2, 3 και προς τα 172.17.17.{1,2,3,4}/32 επιτυγχάνουν, ενώ τα ping προς τα WAN1, 2, 3, 5 αποτυγχάνουν.

7.9

Στο PC1 εκτελούμε ping -R 192.168.2.2. Ακολουθείται η διαδρομή:

```
192.168.1.2 (PC1)      # ICMP request starts
192.168.0.1 (R1)
10.1.1.13  (R4)
192.168.2.1 (R3)
192.168.2.2 (PC2)      # ICMP request is received, ICMP reply starts
10.1.1.14  (R3)
192.168.0.2 (R4)
192.168.1.1 (R1)
192.168.1.2 (PC1)      # ICMP reply is received
```

7.10

Εκτελούμε:

```
### PC1 ###
```

```
traceroute 5.5.5.1          # LAN1 --> 5.5.5.0/24
```

```
### R2 ###
```

```
ping -R -S 172.17.17.2      # 5.5.5.0/24 --> LAN1
```

- Διαδρομή από το LAN1 προς το 5.5.5.0/24

```
0  192.168.1.2 (PC1)
1  192.168.1.1 (R1)
2  10.1.1.5   (R2)
...
64 10.1.1.5   (R2)
```

το ICMP request εγκλωβίζεται στην loopback του R2.

- Διαδρομή από το 5.5.5.0/24 προς το LAN1:

```
172.17.17.2 (R2)
10.1.1.14   (R3)
192.168.0.2 (R4)
192.168.1.1 (R1)
192.168.1.2 (PC1)
```

7.11

Στα PC1,2 εκτελούμε `do ping 10.1.1.9`. Επιβεβαιώνουμε ότι το ping επιτυγχάνει στο PC2, ενώ αποτυγχάνει στο PC1. Αυτό συμβαίνει διότι στον PC2 έχει οριστεί προκαθορισμένη διαδρομή μέσω του R3, οπότε το ICMP request προωθείται στον R3 και από εκεί στην άμεσα συνδεδεμένη διεύθυνση 10.1.1.9 του R1. Αντίθετα, στον PC1 δεν υπάρχει προκαθορισμένη διαδρομή ή σχετική με τη διεύθυνση 10.1.1.9 εγγραφή στον πίνακα δρομολόγησης.

7.12

Στον R2 εκτελούμε `network 0.0.0.0/0`.

7.13

Στον R2 εκτελούμε:

```
do show ip bgp
do show ip route
```

Η προκαθορισμένη διαδρομή έχει προστεθεί στην RIB του R2, παρά το γεγονός ότι δεν έχει προστεθεί στον πίνακα δρομολόγησης, κάτι που δεν είναι αναμενόμενο με βάση τη θεωρία και την λειτουργία του BGP που έχουμε παρατηρήσει έως τώρα στη διάρκεια της άσκησης.

7.14

Στους R1, R3, R4 και PC1 εκτελούμε `do show ip route`. Η προκαθορισμένη διαδρομή έχει προστεθεί στον πίνακα δρομολόγησης των άλλων δρομολογητών και του PC1.

7.15

Στους R1, R3, R4 και PC1 εκτελούμε `do show ip bgp`. Ο τύπος πηγής ORIGIN είναι IGP.

7.16

Στο PC1 εκτελούμε:

```
do ping 10.1.1.2
do ping 10.1.1.6
do ping 10.1.1.10
```

Ναι, μπορούμε να κάνουμε ping στις διευθύνσεις IP και των τριών αυτών WAN.

7.17

Αν στο PC1 εκτελέσουμε `do ping 10.1.1.14` θα λάβουμε μήνυμα λάθους "Destination Host Unreachable" από τον R2. Ακολουθεί η εξήγηση:

- Ο PC1 δεν διαθέτει σχετική εγγραφή για τη διεύθυνση 10.1.1.14, οπότε προωθεί το ICMP request στον R1 μέσω της προκαθορισμένης διαδρομής.

- Ο R1, για τον ίδιο λόγο, προωθεί το ICMP request στον R2 μέσω της προκαθορισμένης διαδρομής.
- Ο R2 δεν διαθέτει ούτε σχετική με την 10.1.1.14 εγγραφή, ούτε προκαθορισμένη διαδρομή, οπότε απαντά με μήνυμα λάθους "Destination Host Unreachable".

7.18

Στον R2 εκτελούμε:

```
no network 0.0.0.0/0
ip route 0.0.0.0/0 172.17.17.2
```

7.19

Στους R1,2,3,4 εκτελούμε `do show ip bgp`. Βλέπουμε ότι πλέον ο τύπος πηγής ORIGIN είναι incomplete.

7.20

Στον R2 εκτελούμε `do show running-config`. Βλέπουμε ότι η εντολή `redistribute static` έχει ήδη δοθεί σε προηγούμενο ερώτημα (συγκεκριμένα στο ερώτημα 3.42).

7.21

Αν στον PC1 εκτελέσουμε `do ping 10.1.1.14` θα λάβουμε μήνυμα λάθους "Time to live exceeded" από τον R2. Αυτό συμβαίνει διότι ο PC1 θα προωθήσει το ICMP request μέσω της προκαθορισμένης διαδρομής στον R1, ο R1 θα το προωθήσει μέσω της προκαθορισμένης διαδρομής στον R2, και ο R2 θα το προωθεί συνεχώς μέσω της προκαθορισμένης διαδρομής στην loopback του, μέχρι να μηδενιστεί το TTL και να σταλεί μήνυμα λάθους TTL exceeded.