ŽILINSKÁ UNIVERZITA V ŽILINE

FAKULTA RIADENIA A INFORMATIKY

Dokumentácia k zadaniu MPLS z predmetu Projektovanie sietí 1

Tomáš Pikna, 5ZKS11

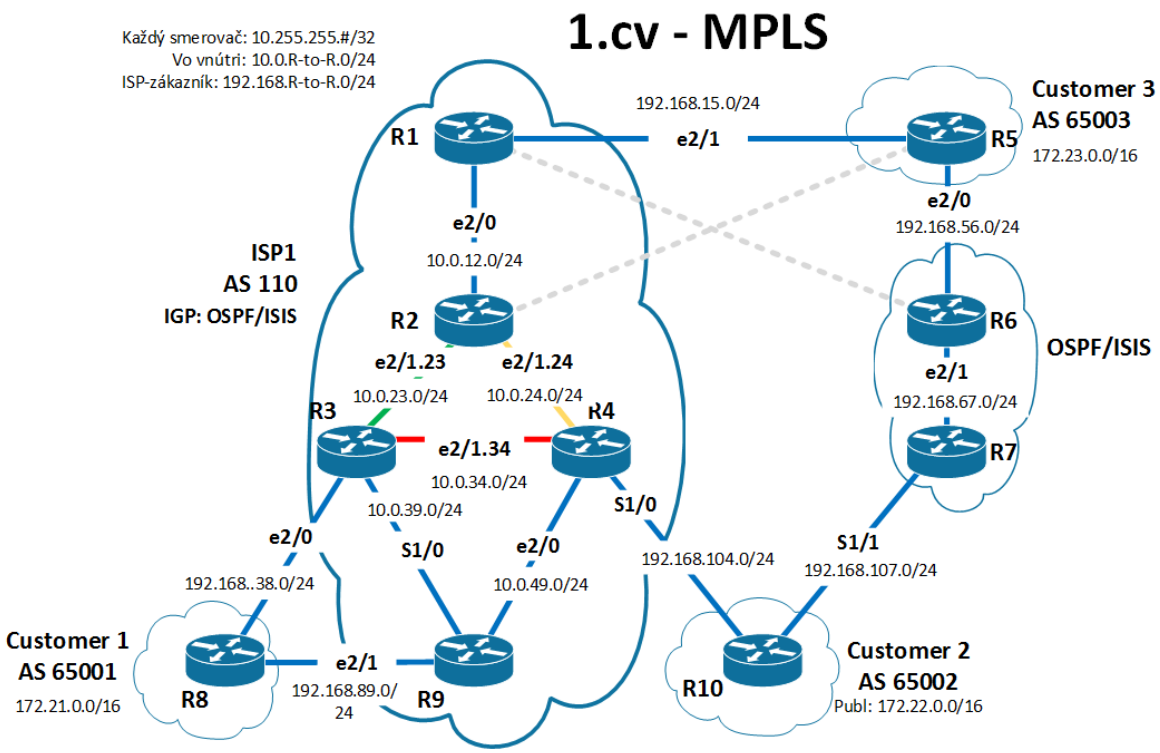
Stanislav Rusnák, 5ZKS11

Akad.rok 2016/2017

1. Zadanie

Vrámci tohto cvičenia sme sa oboznámili s konfiguráciou a fungovaním protokolu MPLS a jeho rôznymi modifikáciami.

2. Topológia



3. Adresovanie

Adresovanie jednotlivých smerovačov a ich rozhraní je uvedené v tabuľke nižšie. Pre rozhranie loopback 0 bola použitá maska /32, pre ostatné rozhrania maska /24.

R1	e2/0	10.0.12.1
	e2/1	192.168.15.1
	lo0	10.255.255.1
R2	e2/0	10.0.12.2
	e2/1.23	10.0.23.2
	e2/1.24	10.0.24.2
	lo0	10.255.255.2
R3	e2/0	192.168.38.3
	e2/1.23	10.0.23.3
	e2/1.34	10.0.34.3
	s1/0	10.0.39.3
	lo0	10.255.255.3
R4	e2/0	10.0.49.4
	e2/1.24	10.0.24.4
	e2/1.34	10.0.34.4
	s1/0	192.168.104.4
	lo0	10.255.255.4
R5	e2/0	192.168.56.5
	e2/1	192.168.15.5
	lo0	10.255.255.5
R6	e2/0	192.168.56.6
	e2/1	192.168.67.6
	lo0	10.255.255.6
R7	e2/0	192.168.67.7
	s1/1	192.168.107.7
	lo0	10.255.255.7
R8	e2/0	192.168.38.8
	e2/1	192.168.89.8
	lo0	10.255.255.8
R9	e2/0	10.0.49.9
	e2/1	192.168.89.9
	s1/0	10.0.39.9
	lo0	10.255.255.9
R10	s1/0	192.168.104.10
	s1/1	192.168.107.10
	lo0	10.255.255.10

4. Použitie IS-IS

Protokol IS-IS bol použitý vrámci ISP1 (siete providera). Konfiguráciu neuvádzame, keďže bola vysvetlená vrámci dokumentácie k IS-IS samotnému. Overenie, že IS-IS je nakonfigurovaný sme vykonali nasledovne :

```
R2#sh run | sec Ethernet2/0
interface Ethernet2/0
 ip address 10.0.12.2 255.255.255.0
 ip router isis
...

R9#sh run | sec Ethernet2/0
interface Ethernet2/0
 ip address 10.0.49.9 255.255.255.0
 ip router isis
...
```

Overili sme si taktiež IS-IS databázu na smerovači R2, kde by mali byť vidieť všetky smerovače na ktorých bol IS-IS pustený.

```
R2#sh isis data

IS-IS Level-2 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
R1.00-00              0x0000029A   0x09FE       583               0/0/0
R2.00-00            * 0x00000294   0xEF9C       1054              0/0/0
R3.00-00              0x00000296   0xB2CD       664               0/0/0
R4.00-00              0x00000290   0x4341       1091              0/0/0
R9.00-00              0x00000290   0x3088       1184              0/0/0
```

5. Konfigurácia MPLS, LDP

Konfigurácia MPLS a LDP sa vykonáva pár základnými príkazmi, ktorými sú :

NA SMEROVAČOCH V GLOBÁLNOM KONFIGURAČNOM MÓDE
ip cef // spustenie Cisco express forwarding
mpls ip //globálne spustenie mpls
mpls label protocol ldp // spustenie protokolu LDP kvôli značkám
NA KAŽDOM ROZHRANÍ KTORÉ MÁ PREPOSIELAŤ MPLS PAKETY MUSÍME SPUSTIŤ MPLS
mpls ip

Nasledujúcim príkazom vynútime použitie Loopbacku0 ako Router-ID kvôli stabilite.

```
mpls ldp router-id Loopback0 force
```

Nasledujúci výpis nám hovorí o MPLS susedoch a nadviazaní spojenia s nimi. Ak je všetko nakonfigurované správne, musí sa v tabuľke nachádzať *xmit/recv.*

```
R1#sh mpls ldp discovery
 Local LDP Identifier:
    10.255.255.1:0 // IP adresa Lo0 R1
    Discovery Sources:
    Interfaces:
       Ethernet2/0 (ldp): xmit/recv //posiela aj príjma
          LDP Id: 10.255.255.2:0 // IP adresa Lo0 R2
```

Vo výpise *sh mpls forwarding-table* môžeme vidieť (L)FIB tabuľku.

```
R1#sh mpls forwarding-table
Local      Outgoing   Prefix          Bytes Label   Outgoing    Next Hop
Label      Label      or Tunnel Id    Switched      interface
16         Pop Label  10.255.255.2/32 0             Fa0/0       10.0.12.2
17         Pop Label  10.0.23.0/24    0             Fa0/0       10.0.12.2
18         Pop Label  10.0.24.0/24    0             Fa0/0       10.0.12.2
19         17         10.255.255.3/32 0             Fa0/0       10.0.12.2
20         18         10.255.255.9/32 0             Fa0/0       10.0.12.2
21         19         10.0.39.0/24    0             Fa0/0       10.0.12.2
22         20         10.0.34.0/24    0             Fa0/0       10.0.12.2
23         21         10.0.49.0/24    0             Fa0/0       10.0.12.2
24         22         10.255.255.4/32 0             Fa0/0       10.0.12.2
```

Aby spolu mohli komunikovať aj siete zákazníka (R5,R8,R10), je potrebné na nich nakonfigurovať BGP protokol voči ich susedom v ISP1 a začať ohlasovať ich svoje siete. Ako príklad poslúži konfigurácia na R8, na zvyšných 2 je konfigurácia podobná, líši sa v označení AS, IP susedov a ohlasovaných sieťach.

```
R8#
router bgp 65001
 bgp log-neighbor-changes
 neighbor 192.168.38.3 remote-as 110
 neighbor 192.168.89.9 remote-as 110

 address-family ipv4
  network 10.255.255.8 mask 255.255.255.255
  neighbor 192.168.38.3 activate
  neighbor 192.168.89.9 activate
```

Ako overenie, že nám spojenie funguje, použijeme traceroute z R5 na ostatné zákaznícke smerovače (R8,R10) kde vidno aj značkovanie trasy.

```
R5#traceroute 10.255.255.8 source 10.255.255.5
Type escape sequence to abort.
Tracing the route to 10.255.255.8

  1 192.168.15.1 [AS 110] 52 msec 40 msec 24 msec
  2 10.0.12.2 [MPLS: Labels 17 Exp 0] 136 msec 128 msec 160 msec
  3 10.0.23.3 [AS 110] 104 msec 76 msec 100 msec
  4 192.168.38.8 [AS 110] 172 msec *  100 msec

R5#traceroute 10.255.255.10 source 10.255.255.5
Type escape sequence to abort.
Tracing the route to 10.255.255.10

  1 192.168.15.1 [AS 110] 28 msec 28 msec 60 msec
  2 10.0.12.2 [MPLS: Labels 22 Exp 0] 120 msec 64 msec 96 msec
  3 10.0.24.4 [AS 110] 72 msec 60 msec 156 msec
  4 192.168.104.10 [AS 110] 100 msec *  112 msec
```

6. Route-reflector

Aby topológia nemusela byť full-mesh, zvolili sme jednoduchší spôsob a to konfiguráciu R2 ako route-reflectora (môže preposielať iBGP informácie naučené cez iBGP).

Pri konfigurácii R2 sme využili možnosť vytvoriť skupinu, do ktorej budú priradení jeho susedia.

```
R2#
router bgp 110
neighbor PEERS peer-group //vytvorenie skupiny PEERS
neighbor PEERS remote-as 110
neighbor PEERS update-source Loopback0
neighbor 10.255.255.1 peer-group PEERS
neighbor 10.255.255.3 peer-group PEERS
neighbor 10.255.255.4 peer-group PEERS
neighbor 10.255.255.9 peer-group PEERS
address-family ipv4 unicast
network 10.255.255.2 mask 255.255.255.255
neighbor PEERS route-reflector-client // konfigurácia R2 ako route-reflector
neighbor 10.255.255.1 activate
neighbor 10.255.255.3 activate
neighbor 10.255.255.4 activate
neighbor 10.255.255.9 activate
```

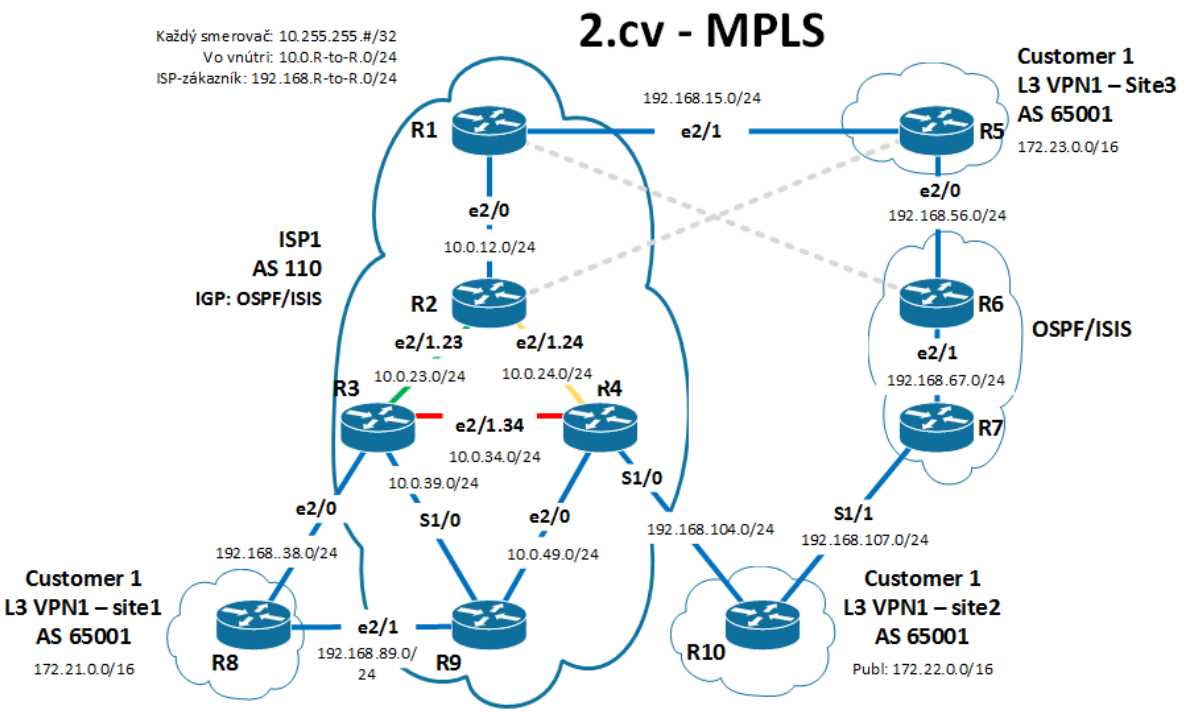
Vrámci AS 110 je potrebné na smerovačoch R1,3,4,9 nadviazať susedstvo s R2. Na každom z týchto smerovačov sme konfigurovali :

```
R1,3,4,9#
router bgp 110
neighbor 10.255.255.2 remote-as 110
neighbor 10.255.255.2 update-source Loopback0
address-family ipv4
neighbor 10.255.255.2 activate
neighbor 10.255.255.2 next-hop-self
network 10.255.255.#R mask 255.255.255.255
```

Dôkazom, že R2 bol nakonfigurovaný korektne je smerovacia tabuľka kde môžeme vidieť rozhrania lo0 všetkých smerovačov (R6 a R7 sa nepoužívajú).

```
R1#sh ip route
C 192.168.15.0/24 is directly connected, FastEthernet0/1
10.0.0.0/8 is variably subnetted, 14 subnets, 2 masks
B 10.255.255.8/32 [200/0] via 10.255.255.3, 01:38:54
B 10.255.255.10/32 [200/0] via 10.255.255.4, 01:42:35
i L1 10.255.255.9/32 [115/40] via 10.0.12.2, FastEthernet0/0
C 10.0.12.0/24 is directly connected, FastEthernet0/0
i L1 10.255.255.2/32 [115/20] via 10.0.12.2, FastEthernet0/0
i L1 10.255.255.3/32 [115/30] via 10.0.12.2, FastEthernet0/0
C 10.255.255.1/32 is directly connected, Loopback0
i L1 10.255.255.4/32 [115/30] via 10.0.12.2, FastEthernet0/0
B 10.255.255.5/32 [20/0] via 192.168.15.5, 01:46:48
i L1 10.0.24.0/24 [115/20] via 10.0.12.2, FastEthernet0/0
i L1 10.0.23.0/24 [115/20] via 10.0.12.2, FastEthernet0/0
i L1 10.0.34.0/24 [115/30] via 10.0.12.2, FastEthernet0/0
i L1 10.0.39.0/24 [115/30] via 10.0.12.2, FastEthernet0/0
i L1 10.0.49.0/24 [115/30] via 10.0.12.2, FastEthernet0/0
```

7. Topológia – cvičenie 2



V tomto cvičení nastali zmeny v topológii a to :

- Zákazníci v jednom AS 65001
- Nové siete zákazníkov na rozhraní Loopback10
    - R5 : 172.23.5.0 /24
    - R8 : 172.21.8.0 /24
    - R10 : 172.22.10.0 /24

Kvôli zmene AS bolo potrebné na R5,8,10 vypnúť pôvodné BGP.

Okrajové smerovače providera označujeme ako PE. Na nich bolo potrebné zapnúť VRF pre zákazníka 1. Aby tieto cesty boli preňho unikátne bolo taktiež potrebné definovať route distinguisher (rd) a route-target.

```
R1,3,4,9#
ip vrf z1
rd 110:1
route-target 110:1
```

Na všetky rozhrania PE smerovačov, ktoré smerujú k zákazníkom bolo potrebné zadať príkaz:

```
ip vrf forwarding z1
```

Po zadaní tohto príkazu sa IP adresa z rozhrania zmaže a je potrebné ju zadať nanovo. Zároveň sa týmto príkazom presunie záznam z globálnej smerovacej tabuľky do tabuľky vrf z1. Ak chceme zistiť, ktoré rozhrania sú vo vrf z1, použijeme príkaz:

```
R3#sh ip vrf
  Name                 Default RD        Interfaces
  z1                   110:1             Et2/0
```

V tomto cvičení sme sa rozhodli, že route-reflectorom bude smerovač R1. Preto boli potrebné zmeny v konfigurácii a to :

```
R1#
router bgp 110
no bgp default ipv4-unicast
neighbor 10.255.255.3 remote-as 110
neighbor 10.255.255.3 update-source Loopback0
neighbor 10.255.255.4 remote-as 110
neighbor 10.255.255.4 update-source Loopback0
neighbor 10.255.255.9 remote-as 110
neighbor 10.255.255.9 update-source Loopback0
address-family vpnv4
neighbor 10.255.255.3 activate
neighbor 10.255.255.3 route-reflector-client
neighbor 10.255.255.4 activate
neighbor 10.255.255.4 route-reflector-client
neighbor 10.255.255.9 activate
neighbor 10.255.255.9 route-reflector-client
```

Pre ostatné PE smerovače bola konfigurácia nasledovná:

```
R3,4,9#
router bgp 110
no bgp default ipv4-unicast
neighbor 10.255.255.1 remote-as 110
neighbor 10.255.255.1 update-source Lo0
address-family vpnv4
neighbor 10.255.255.1 activate
```

Aby sa distribuovali v BGP aj pripojené siete zákazníka, bolo potrebné nakonfigurovať na PE smerovačoch :

```
R1,3,4,9#
router bgp 110
address-family ipv4 vrf z1
```

```
redistribute connected
```

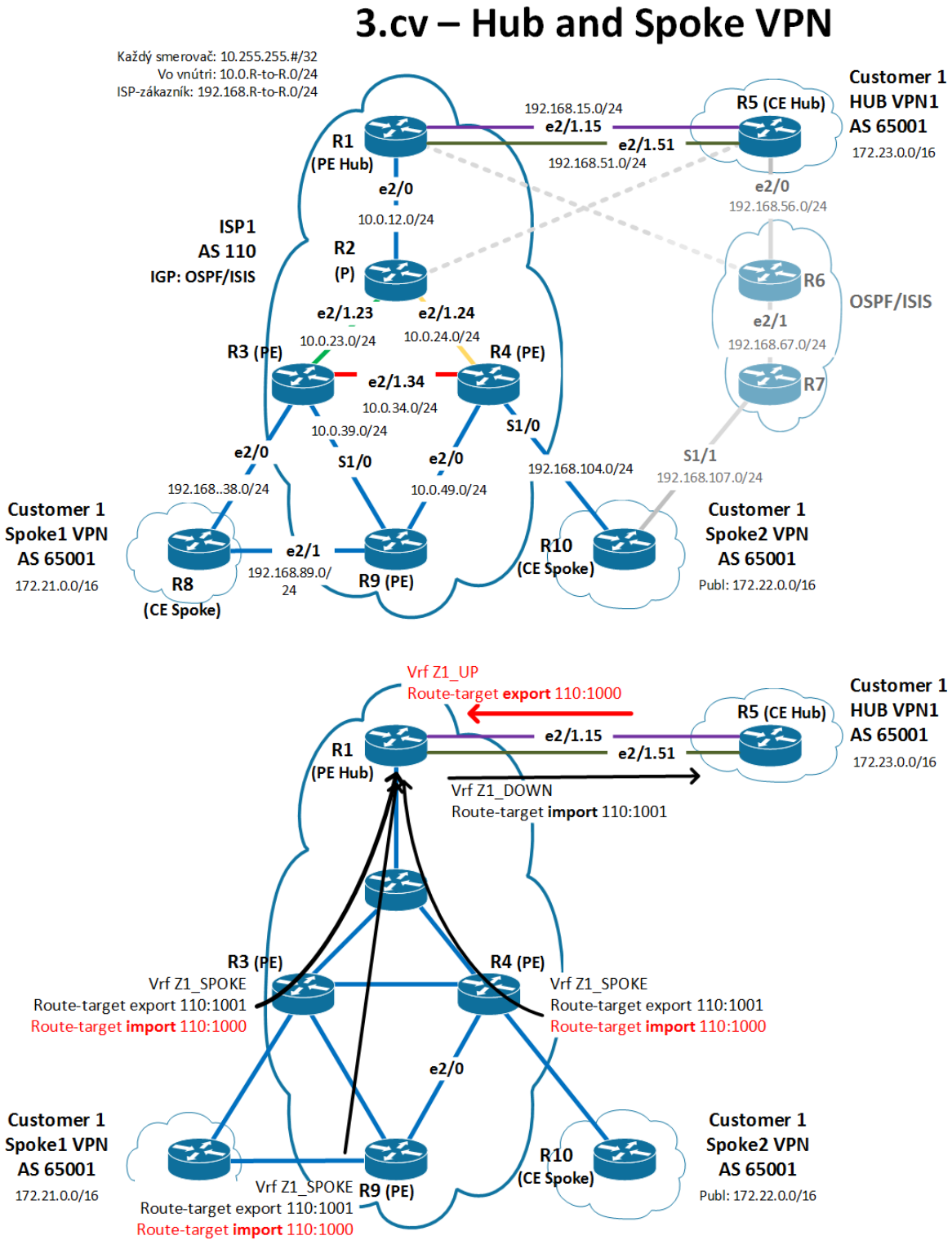
Je potrebné nadviazať aj BGP spojenie medzi PE a CE (Customer Edge) smerovačmi. Príkazom *as-override* zabezpečíme, aby CE smerovače nezahadzovali updaty zo svojho AS 65001. Ako príklad nám poslúžia smerovače R3 a R8. V R3 musíme suseda pridať do *address-family vrf z1* a pri R8 je to v *ipv4 unicast* kde ohlasujeme aj pripojené siete.

```
R3#
router bgp 110
address-family ipv4 vrf z1
neighbor 192.168.38.8 remote-as 65001
neighbor 192.168.38.8 activate
neighbor 192.168.38.8 as-override

R8#
router bgp 65001
neighbor 192.168.38.3 remote-as 110
neighbor 192.168.89.9 remote-as 110
address-family ipv4
network 10.255.255.8 mask 255.255.255.255
network 172.21.8.0 mask 255.255.255.0
neighbor 192.168.38.3 activate
neighbor 192.168.89.9 activate
```

Ako overenie dobrej konfigurácie môžeme použiť smerovaciu tabuľku *ipv4 unicast*, kde by sme mali vidieť všetky zákaznícke siete z R5, R8, R10.

```
R5#sh ip bgp ipv4 unicast

     Network          Next Hop            Metric LocPrf Weight Path
 *>  10.255.255.5/32  0.0.0.0                  0         32768 i
 *>  10.255.255.8/32  192.168.15.1                         0 110 110 i
 *>  10.255.255.10/32 192.168.15.1                         0 110 110 i
 *>  172.21.8.0/24    192.168.15.1                         0 110 110 i
 *>  172.22.10.0/24   192.168.15.1                         0 110 110 i
 *>  172.23.5.0/24    0.0.0.0                  0         32768 i
 r>  192.168.15.0     192.168.15.1             0             0 110 ?
 *>  192.168.38.0     192.168.15.1                         0 110 ?
 *>  192.168.89.0     192.168.15.1                         0 110 ?
 *>  192.168.104.0    192.168.15.1                         0 110 ?
```

8. Topológia – cvičenie 3 – Hub-and-spoke



Úlohou v tomto cvičení bolo nakonfigurovať Hub-and-spoke technológiu, kedy všetky zákaznícke siete posielajú svoj traffic na centrálny uzol (Hub). To z dôvodu jednoduchšej administrácie jednotlivých VPN ako napríklad použitie firewallov a podobne. Zákaznícke siete posielajú svoje routy na hub, ten ich iným fyzickým alebo logickým portom pošle do ostatných zákazníckych sietí s iným route-targetom aby sa k nim dostali.

V prvom rade bolo potrebné zrušiť *vrf z1*. Následne na PE smerovačoch R3,R4 a R9 vytvoriť *vrf z1_spoke* s ktorá bude exportovať svoje cesty s route-targetom 110:1000 a importovať cesty ktoré sú značené route-targetom 110:1001.

```
R3,4,9#
ip vrf z1_spoke
rd 110:1
route-target export 110:1000
route-target import 110:1001
```

a na ich rozhraniach voči CE smerovačom zapnúť forwarding tejto vrf a nanovo nastaviť IP adresy:

```
R3,4,9#
ip vrf forwarding z1_spoke
```

Na smerovači R1 a R5 bolo potrebné rozdeliť rozhranie Ethernet2/1 na 2 logické celky e2/1.15 (z R1 na R5)a e2/1.51 (z R5 na R1). Bolo tu potrebné vytvoriť aj 2 vrf (na R1):

- vrf z1_up – smer od hubu k zákazníkom
- vrf z1_down – smer od zákazníkov k hubu

```
R1#
ip vrf z1_up
ip route vrf z1_up 0.0.0.0 0.0.0.0 192.168.51.5
rd 110:2
route-target export 110:1001

router bgp 110
address-family ipv4 vrf z1_up
redistribute static
neighbor 192.168.51.5 remote-as 65001
neighbor 192.168.51.5 activate
default-information originate
```

```
R1#
ip vrf z1_down
rd 110:1
route-target import 110:1000

router bgp 110
address-family ipv4 vrf z1_down
neighbor 192.168.15.5 remote-as 65001
neighbor 192.168.15.5 activate
neighbor 192.168.15.5 as-override
```

Taktiež na rozhraniach bolo potrebné nakonfigurovať forwardovanie jednotlivých vrf + znovunastavenie IP adries:

```
R1#
interface Ethernet2/1.15
encapsulation dot1Q 15
ip vrf forwarding z1_down
..

interface Ethernet2/1.51
encapsulation dot1Q 51
ip vrf forwarding z1_up
..
```

Na nasleduúcich výpisoch môžeme vidieť výpisy tabuliek z R1,R8 a R9 pre ipv4 unicast a pre vrf z1_spoke, z1_up a z1_down.

```
R8#sh ip bgp ipv4 unicast

   Network          Next Hop            Metric LocPrf Weight Path
*   0.0.0.0          192.168.89.9                           0 110 ?
*>                   192.168.38.3                           0 110 ?
*>  10.255.255.8/32  0.0.0.0                  0         32768 i
*>  172.21.8.0/24    0.0.0.0                  0         32768 i

R9#sh ip bgp vpnv4 vrf z1_spoke

   Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 110:1 (default for vrf z1_spoke)
*>i 0.0.0.0          10.255.255.1             0    100      0 ?
*>  10.255.255.8/32  192.168.89.8             0             0 65001 i
*>  172.21.8.0/24    192.168.89.8             0             0 65001 i

R1#sh ip bgp vpnv4 vrf z1_up

   Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 110:2 (default for vrf z1_up)
*>  0.0.0.0          192.168.51.5             0         32768 ?

------------------------------------------------------------------

R1#sh ip bgp vpnv4 vrf z1_down

   Network          Next Hop            Metric LocPrf Weight Path
```

```
Route Distinguisher: 110:1 (default for vrf z1_down)
 *>  10.255.255.5/32  192.168.15.5          0          0 65001 i
 * i 10.255.255.8/32  10.255.255.9          0    100     0 65001 i
 *>i                  10.255.255.3          0    100     0 65001 i
 *>i 10.255.255.10/32 10.255.255.4          0    100     0 65001 i
 * i 172.21.8.0/24    10.255.255.9          0    100     0 65001 i
 *>i                  10.255.255.3          0    100     0 65001 i
 *>i 172.22.10.0/24   10.255.255.4          0    100     0 65001 i
 *>  172.23.5.0/24    192.168.15.5          0          0 65001 i
```

Ako dôkaz nám môžu poslúžiť aj nasledujúce tracerouty, kde prevádzka ide cez HUB :

```
R10#traceroute 10.255.255.8 source 10.255.255.10
Type escape sequence to abort.
Tracing the route to 10.255.255.8
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.104.4 [AS 110] 32 msec 20 msec 16 msec
  2 10.0.24.2 [AS 110] [MPLS: Labels 19/27 Exp 0] 84 msec 76 msec 80 msec
  3 192.168.51.1 [AS 110] [MPLS: Label 27 Exp 0] 80 msec 80 msec 80 msec
  4 192.168.51.5 [AS 110] 80 msec 80 msec 80 msec
  5 192.168.15.1 [AS 110] 84 msec 80 msec 80 msec
  6  *  *
    10.0.12.2 [AS 110] [MPLS: Labels 20/24 Exp 0] 152 msec
  7 192.168.38.3 [AS 110] [MPLS: Label 24 Exp 0] 152 msec 168 msec 148 msec
  8 192.168.38.8 [AS 110] 156 msec *  196 msec

R8#traceroute 10.255.255.10 source 10.255.255.8
Type escape sequence to abort.
Tracing the route to 10.255.255.10
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.38.3 [AS 110] 36 msec 24 msec 20 msec
  2 10.0.23.2 [AS 110] [MPLS: Labels 19/27 Exp 0] 76 msec 80 msec 84 msec
  3 192.168.51.1 [AS 110] [MPLS: Label 27 Exp 0] 80 msec 56 msec 104 msec
  4 192.168.51.5 [AS 110] 80 msec 56 msec 100 msec
  5 192.168.15.1 [AS 110] 80 msec 60 msec 92 msec
  6  *  *  *
  7 192.168.104.4 [AS 110] [MPLS: Label 25 Exp 0] 152 msec 116 msec 164 msec
  8 192.168.104.10 [AS 110] 148 msec *  176 msec
```

9. Draft rosen

Pre potreby tohto cvičenia sme vychádzali z topológie a konfigurácie, ktorú sme mali na cvičení č.2. Draft rosen je označenie technológie, kedy sa využíva multicast cez VPN.

Ako prvé bolo potrebné na všetkých smerovačoch zapnúť podporu multicastov príkazom

```
ip multicast-routing
```

Zároveň je potrebné na všetkých rozhraniach, vrátane Loopbackov, ktoré budeme neskôr pripájať do multicastovej skupiny zadať príkaz:

```
ip pim sparse-mode
```

Na PE smerovačoch (R1,3,4,9) je potrebné povoliť taktiež multicast pre vrf z1 príkazom:

```
ip multicast-routing vrf z1
```

Je potrebné nakonfigurovať Randezvous point aj pre providerovu aj pre zákaznícku sieť zvlášť, pretože zákaznícka multicastová sieť nevie nič o providerovej multicastovej sieti.

Pre providerovu sieť sme konfigurovali RP na R1,2,3,4,9. Ako RP sme zvolili R1 príkazom:

```
ip pim rp-address 10.255.255.1
```

Pre zákaznícku sieť sme zvolili ako RP smerovač R5 a konfigurovali sme to na PE smerovačoch R1,3,4,9. Taktiež je potrebné nakonfigurovať MDT (Multicast Distribution Tree), pomocou ktorého PE smerovače vytvárajú GRE tunely a zákaznícky traffic bude následne do týchto GRE paketov enkapsulovaný. Použili sme na to tieto príkazy:

```
ip pim vrf z1 rp-address 172.23.5.5
ip vrf z1
mdt default 239.0.0.1
```

Aby sme mohli konfiguráciu otestovať, na smerovačoch R8 a R10 sme rozhrania loopback10 pripojili do multicastovej skupiny príkazom:

```
int lo10
ip igmp join-group 239.1.1.1
```

Zo zákazníckych smerovačov R5,8,10 sme príkazom ping na multicastovú adresu 239.1.1.1 testovali funkčnosť.

```
R5(config-if)#do ping 239.1.1.1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 239.1.1.1, timeout is 2 seconds:
```

```
Reply to request 0 from 172.22.10.10, 88 ms
Reply to request 0 from 172.21.8.8, 92 ms
```

```
R8(config-if)#do ping 239.1.1.1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 239.1.1.1, timeout is 2 seconds:

Reply to request 0 from 172.21.8.8, 48 ms
Reply to request 0 from 172.22.10.10, 220 ms
```

```
R10(config-if)#do ping 239.1.1.1
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 239.1.1.1, timeout is 2 seconds:

Reply to request 0 from 172.22.10.10, 64 ms
Reply to request 0 from 172.21.8.8, 336 ms
```