Exercício 2 – iBGP

Objetivo: Configurar as sessões iBGP com o intuito de repassar as rotas aprendidas externamente para todos os roteadores do AS e também para divulgar as rotas para as redes dos clientes.

Cenário inicial: Protocolo de roteamento interno configurado (OSPFv2 e OSPFv3).

1. Primeiro, crie uma interface de *loopback* em cada roteador que será utilizada para estabelecer as sessões iBGP.

   No roteador mikrotik_borda utilize os seguintes comandos:

```
> interface bridge add name=lo20 auto-mac=no admin-mac=1A:B0:06:01:XX:20
> ip address add address=102.XX.0.254/32 interface=lo20 comment=IBGP
> ipv6 address add address=4D0C:XX::254/128 interface=lo20 comment=IBGP
```

   No roteador mikrotik_clientes utilize os seguintes comandos:

```
> interface bridge add name=lo20 auto-mac=no admin-mac=1A:B0:06:02:XX:20
> ip address add address=102.XX.0.251/32 interface=lo20 comment=IBGP
> ipv6 address add address=4D0C:XX::251/128 interface=lo20 comment=IBGP
```

   No roteador cisco utilize os seguintes comandos:

```
# configure terminal
# interface Loopback20
# description IBGP
# ip address 102.XX.16.254 255.255.255.255
# ipv6 address 4D0C:XX:8000::254:1/112
# exit
# exit
```

No roteador juniper, como ele não suporta mais de uma interface de *loopback*, utilize endereço que foi configurado anteriormente.

2. O passo seguinte é adicionar as interfaces de *loopback* criadas nos mikrotiks e no cisco à área 0 do OSPFv2 e do OSPFv3.

   No roteador mikrotik_borda utilize os seguintes comandos:

```
> routing ospf network add area=backbone disabled=no network=102.XX.0.254/32
> routing ospf-v3 interface add area=backbone interface=lo20 passive=yes
```

   No roteador mikrotik_clientes utilize os seguintes comandos:

```
> routing ospf network add area=backbone disabled=no network=102.XX.0.251/32
> routing ospf-v3 interface add area=backbone interface=lo20 passive=yes
```

No roteador cisco utilize os seguintes comandos:

```
# configure terminal
# interface Loopback20
# ip ospf 100 area 0
# ipv6 ospf 200 area 0
# ipv6 ospf network point-to-point
# router ospf 100
# passive-interface Loopback20
# ipv6 router ospf 200
# passive-interface Loopback20
# exit
# exit
```

3. Teste a conectividade entre as *loopback* criadas, tanto IPv4 quanto IPv6.

4. No passo seguinte, configure as sessões iBGP entre os roteadores. Com isso, quando as sessões eBGP com os provedores de trânsito estiverem configuradas, a rede do AS estará preparada para divulgar as redes dos clientes e propagar internamente o que for aprendido da Internet. A configuração será feita em *full mesh*, ou seja, todos os roteadores terão sessões iBGP com os demais dentro do próprio AS.

   No roteador mikrotik_borda utilize os seguintes comandos:

```
> /routing bgp instance set default as=655XX router-id=102.XX.0.255
> /routing bgp peer
  add address-families=ip name=IBGP-IPV4-MKT-CLIENTES remote-as=655XX \
nexthop-choice=force-self remote-address=102.XX.0.251 update-source=lo20

  add address-families=ip name=IBGP-IPV4-CISCO nexthop-choice=force-self \
remote-address=102.XX.16.254 remote-as=655XX  update-source=lo20

  add address-families=ip name=IBGP-IPV4-JUNIPER nexthop-choice=force-self \
remote-address=102.XX.16.252 remote-as=655XX update-source=lo20

  add address-families=ipv6 name=IBGP-IPV6-CISCO update-source=lo20 \
remote-as=655XX nexthop-choice=force-self remote-address=4D0C:XX:8000::254:1

  add address-families=ipv6 name=IBGP-IPV6-JUNIPER update-source=lo20 \
nexthop-choice=force-self remote-address=4D0C:XX:8000::252:1 remote-as=655XX

  add address-families=ipv6 name=IBGP-IPV6-MKT-CLIENTES update-source=lo20 \
nexthop-choice=force-self remote-address=4D0C:XX::251 remote-as=655XX
> /
```

No roteador mikrotik_clientes utilize os seguintes comandos:

```
> /routing bgp instance set default as=655XX router-id=102.XX.0.252
> /routing bgp peer
  add address-families=ip name=IBGP-IPV4-MKT-BORDA update-source=lo20 \
nexthop-choice=force-self remote-address=102.XX.0.254 remote-as=655XX
  add address-families=ip name=IBGP-IPV4-CISCO nexthop-choice=force-self \
remote-address=102.XX.16.254 remote-as=655XX  update-source=lo20
  add address-families=ip name=IBGP-IPV4-JUNIPER nexthop-choice=force-self \
remote-address=102.XX.16.252 remote-as=655XX update-source=lo20

  add address-families=ipv6 name=IBGP-IPV6-CISCO nexthop-choice=force-self \
remote-address=4D0C:XX:8000::254:1 remote-as=655XX update-source=lo20
  add address-families=ipv6 name=IBGP-IPV6-JUNIPER update-source=lo20 \
nexthop-choice=force-self remote-address=4D0C:XX:8000::252:1 remote-as=655XX
  add address-families=ipv6 name=IBGP-IPV6-MKT-BORDA update-source=lo20 \
nexthop-choice=force-self remote-address=4D0C:XX::254 remote-as=655XX
> /
```

No roteador cisco utilize os seguintes comandos:

```
# configure terminal
# router bgp 655XX
# bgp router-id 102.XX.16.255
# no bgp default ipv4-unicast
# bgp log-neighbor-changes
# neighbor 4D0C:XX::254 remote-as 655XX
# neighbor 4D0C:XX::254 description IBGP-IPV6-MKT-BORDA
# neighbor 4D0C:XX::254 update-source Loopback20
# neighbor 4D0C:XX::251 remote-as 655XX
# neighbor 4D0C:XX::251 description IBGP-IPV6-MKT-CLIENTES
# neighbor 4D0C:XX::251 update-source Loopback20
# neighbor 4D0C:XX:8000::252:1 remote-as 655XX
# neighbor 4D0C:XX:8000::252:1 description IBGP-IPV6-JUNIPER
# neighbor 4D0C:XX:8000::252:1 update-source Loopback20

# neighbor 102.XX.0.254 remote-as 655XX
# neighbor 102.XX.0.254 description IBGP-IPV4-MKT-BORDA
# neighbor 102.XX.0.254 update-source Loopback20
# neighbor 102.XX.0.251 remote-as 655XX
# neighbor 102.XX.0.251 description IBGP-IPV4-MKT-CLIENTES
# neighbor 102.XX.0.251 update-source Loopback20
# neighbor 102.XX.16.252 remote-as 655XX
# neighbor 102.XX.16.252 description IBGP-IPV4-JUNIPER
# neighbor 102.XX.16.252 update-source Loopback20

# address-family ipv6
# neighbor 4D0C:XX::254 activate
# neighbor 4D0C:XX::254 next-hop-self
# neighbor 4D0C:XX::254 soft-reconfiguration inbound
# neighbor 4D0C:XX::251 activate
# neighbor 4D0C:XX::251 next-hop-self
# neighbor 4D0C:XX::251 soft-reconfiguration inbound
# neighbor 4D0C:XX:8000::252:1 activate
# neighbor 4D0C:XX:8000::252:1 next-hop-self
# neighbor 4D0C:XX:8000::252:1 soft-reconfiguration inbound

# address-family ipv4
# neighbor 102.XX.0.254 activate
# neighbor 102.XX.0.254 next-hop-self
# neighbor 102.XX.0.254 soft-reconfiguration inbound
# neighbor 102.XX.0.251 activate
# neighbor 102.XX.0.251 next-hop-self
# neighbor 102.XX.0.251 soft-reconfiguration inbound
```

```
# neighbor 102.XX.16.252 activate
# neighbor 102.XX.16.252 next-hop-self
# neighbor 102.XX.16.252 soft-reconfiguration inbound
# exit
# exit
# exit
# copy running-config startup-config
```

No roteador juniper utilize os seguintes comandos:

```
> edit
# set routing-options autonomous-system 655XX
# set protocols bgp group IBGP-IPV4 type internal
# set protocols bgp group IBGP-IPV4 local-address 102.XX.16.252
# set protocols bgp group IBGP-IPV4 neighbor 102.XX.0.254 description IBGP-
IPV4-MKT-BORDA
# set protocols bgp group IBGP-IPV4 neighbor 102.XX.0.251 description IBGP-
IPV4-MKT-CLIENTES
# set protocols bgp group IBGP-IPV4 neighbor 102.XX.16.254 description IBGP-
IPV4-CISCO

# set protocols bgp group IBGP-IPV6 type internal
# set protocols bgp group IBGP-IPV6 local-address 4D0C:XX:8000::252:1
# set protocols bgp group IBGP-IPV6 neighbor 4D0C:XX::254 description IBGP-
IPV6-MKT-BORDA
# set protocols bgp group IBGP-IPV6 neighbor 4D0C:XX::251 description IBGP-
IPV6-MKT-CLIENTES
# set protocols bgp group IBGP-IPV6 neighbor 4D0C:XX:8000::254:1 description
IBGP-IPV6-CISCO
# commit
```

5. Para verificar se todas as sessões foram estabelecidas corretamente, utilize os comandos listados abaixo. Atente ao estado e à quantidade de prefixos aprendidos de cada vizinho.

   Nos roteadores mikrotik_borda e mikrotik_clientes, utilize o seguinte comando:

```
> routing bgp peer print detail status
```

   No roteador cisco utilize os seguintes comandos:

```
# show bgp ipv4 unicast summary
# show bgp ipv6 unicast summary
```

   No roteador juniper utilize o seguinte comando:

```
> show bgp summary
```

Com as sessões estabelecidas, podemos agora configurar o iBGP para divulgar as redes dos clientes. Isso será feito através da criação de políticas de roteamento no juniper e da declaração das redes através do comando `network` no mikrotik_clientes.

No roteador juniper os comandos a serem utilizados são os seguintes:

```
> edit
# set policy-options policy-statement NETWORK-CLIENTES-IPV4 term 1 from
route-filter 102.XX.17.0/28 exact
```

```
# set policy-options policy-statement NETWORK-CLIENTES-IPV4 term 1 from
route-filter 102.XX.24.0/21 exact
# set policy-options policy-statement NETWORK-CLIENTES-IPV4 term 1 then
next-hop self
# set policy-options policy-statement NETWORK-CLIENTES-IPV4 term 1 then
accept
# set policy-options policy-statement NETWORK-CLIENTES-IPV4 term implicit-
deny then reject

# set policy-options policy-statement NETWORK-CLIENTES-IPV6 term 1 from
route-filter 4D0C:XX:8001::/48 exact
# set policy-options policy-statement NETWORK-CLIENTES-IPV6 term 1 from
route-filter 4D0C:XX:8800::/40 exact
# set policy-options policy-statement NETWORK-CLIENTES-IPV6 term 1 then
next-hop self
# set policy-options policy-statement NETWORK-CLIENTES-IPV6 term 1 then
accept
# set policy-options policy-statement NETWORK-CLIENTES-IPV6 term implicit-
deny then reject

# set protocols bgp group IBGP-IPV4 neighbor 102.XX.0.254 export NETWORK-
CLIENTES-IPV4
# set protocols bgp group IBGP-IPV4 neighbor 102.XX.0.251 export NETWORK-
CLIENTES-IPV4
# set protocols bgp group IBGP-IPV4 neighbor 102.XX.16.254 export NETWORK-
CLIENTES-IPV4

# set protocols bgp group IBGP-IPV6 neighbor 4D0C:XX::254 export NETWORK-
CLIENTES-IPV6
# set protocols bgp group IBGP-IPV6 neighbor 4D0C:XX::251 export NETWORK-
CLIENTES-IPV6
# set protocols bgp group IBGP-IPV6 neighbor 4D0C:XX:8000::254:1 export
NETWORK-CLIENTES-IPV6

# set routing-options static route 102.XX.17.0/28 next-hop 102.XX.17.2
passive
# set routing-options static route 102.XX.24.0/21 next-hop 102.XX.24.1
passive

# set routing-options rib inet6.0 static route 4D0C:XX:8001::/48 next-hop
4D0C:XX:8001::2 passive
# set routing-options rib inet6.0 static route 4D0C:XX:8800::/40 next-hop
4D0C:XX:8800::2 passive

# commit
```

No roteador mikrotik_clientes utilize os seguintes comandos:

```
/routing bgp network
  add disabled=no network=102.XX.1.0/28 synchronize=no
  add disabled=no network=102.XX.2.0/30 synchronize=no
  add disabled=no network=102.XX.8.0/21 synchronize=no
  add disabled=no network=4D0C:XX:1::/48 synchronize=no
  add disabled=no network=4D0C:XX:2::/126 synchronize=no
  add disabled=no network=4D0C:XX:800::/40 synchronize=no
```

Nos roteadores mikrotik, todas as rotas foram consideradas válidas?